Tencent Cloud TI-ONE Platform Operation Guide



©2013-2026 Tencent Cloud. All rights reserved.

Copyright Notice

©2013-2026 Tencent Cloud. All rights reserved.

The complete copyright of this document, including all text, data, images, and other content, is solely and exclusively owned by Tencent Cloud Computing (Beijing) Co., Ltd. ("Tencent Cloud"); Without prior explicit written permission from Tencent Cloud, no entity shall reproduce, modify, use, plagiarize, or disseminate the entire or partial content of this document in any form. Such actions constitute an infringement of Tencent Cloud's copyright, and Tencent Cloud will take legal measures to pursue liability under the applicable laws.

Trademark Notice

Tencent Cloud

This trademark and its related service trademarks are owned by Tencent Cloud Computing (Beijing) Co., Ltd. and its affiliated companies("Tencent Cloud"). The trademarks of third parties mentioned in this document are the property of their respective owners under the applicable laws. Without the written permission of Tencent Cloud and the relevant trademark rights owners, no entity shall use, reproduce, modify, disseminate, or copy the trademarks as mentioned above in any way. Any such actions will constitute an infringement of Tencent Cloud's and the relevant owners' trademark rights, and Tencent Cloud will take legal measures to pursue liability under the applicable laws.

Service Notice

This document provides an overview of the as-is details of Tencent Cloud's products and services in their entirety or part. The descriptions of certain products and services may be subject to adjustments from time to time.

The commercial contract concluded by you and Tencent Cloud will provide the specific types of Tencent Cloud products and services you purchase and the service standards. Unless otherwise agreed upon by both parties, Tencent Cloud does not make any explicit or implied commitments or warranties regarding the content of this document.

Contact Us

We are committed to providing personalized pre-sales consultation and technical after-sale support. Don't hesitate to contact us at 4009100100 or 95716 for any inquiries or concerns.

Contents

Operation Guide
- Model Hub
- Data Center
  - Data Center Introduction
  - Creating a Dataset
  - Dataset List Operations
  - Creating a Data Labeling Task for an LLM
  - Creating a Data Labeling Task for Traditional CV
  - Operations on the Data Labeling List
  - Labeling Workbench for Traditional CV Data
  - Tag Format Examples
  - Data Construction
- Task-based Modeling
  - Overview of Task-based Modeling
  - Creating a Task
  - Task Management
  - User Guide on Distributed Training
  - Guide on Configuration of Task Event Alarms
  - Data Format Specifications for Training Built-in LLMs
- Dev Machines
  - Introduction to Dev Machines
  - Creating a Dev Machine
  - Managing Dev Machines
  - Guide on Configuring an Idle Recycling Policy
  - Using Lifecycle Scripts
  - Using the Podman Image Tool in a Dev Machine
- Model Management
  - Introduction to Model Management
  - Usage of Model Repository
  - Usage of Model Optimization
  - Model Package Specifications
- Model Evaluation
  - Introduction to Model Evaluation
  - Quick Test
  - Automated Evaluation
  - Manual Evaluation

©2013-2026 Tencent Cloud. All rights reserved.

Evaluation-based Visual Comparison
Evaluation Set Format Requirements
Automated Evaluation: Prompt for Judge Model Scoring and Preprocessing/Postprocessing Format Requirements
Automated Evaluation: Quick Configuration File Specifications for Metrics
Introduction to Built-in Open-Source Evaluation Sets
Online Services
Introduction to Online Services
Deployment of Online Services
Authentication and Traffic Throttling for Online Services
Operations of Online Services
Guide on Releasing Online Services Using the Built-in Images of the Platform
Guide on Releasing Online Services Using Custom Images
Model Inference File Introduction and Examples
Batch Prediction
Introduction to Batch Prediction
Platform Management
Data Source Management
Enterprise Management
Resource Group Management
Resource Group Introduction
Configuration of the Node Network
Scheduling and Quota Management
File Preloading
Fault Diagnosis and Recovery
Resource Monitor
Introduction to Resource Monitor
Metric Description
GPU Virtualization
Tikit
Tikit Introduction
Installation and Initialization
Viewing Resources in a Resource Group
Uploading and Downloading Data
Submitting a Training Task
Creating and Managing Dev Machines
Registering a Model
Model Optimization
Submitting a Batch Task

©2013-2026 Tencent Cloud. All rights reserved.

Operation Guide
Model Hub

Last updated: 2026-02-25 11:43:34

Overview

Model Hub is a built-in LLM library of Tencent Cloud TI-ONE Platform (TI-ONE). It is preset with a variety of pre-trained LLMs and instruction-tuned LLMs that cover a wide range of downstream tasks, including multi-turn chats, logical inference, and content creation.
You can easily deploy the built-in LLMs as online services with a single click, and therefore quickly and intuitively experience the effectiveness of the LLMs through web-based Q&A. Additionally, you can initiate fine-tuning tasks based on the built-in models, optimizing the models to meet production-level requirements for specific vertical scenarios.

Built-in LLMs

The following LLMs have been built into Model Hub. The latest list of supported models is subject to the one displayed on the Model Hub page.

- Self-developed Tencent Hunyuan: including the open-source Hunyuan-Large series, and the closed-source versions.

  Featured general-purpose LLMs: including mainstream open-source LLMs, such as the DeepSeek series, Qwen3 series, Qwen3-VL series, Qwen2.5-VL series, Gemma 3 series, Kimi-K2 series.
- Others: including Tencent self-developed industry-specific LLMs.

For access to the closed-source versions of Tencent Hunyuan and the industry-specific LLMs mentioned above, contact Tencent Cloud's industry sales team to enable the services.

Feature Description

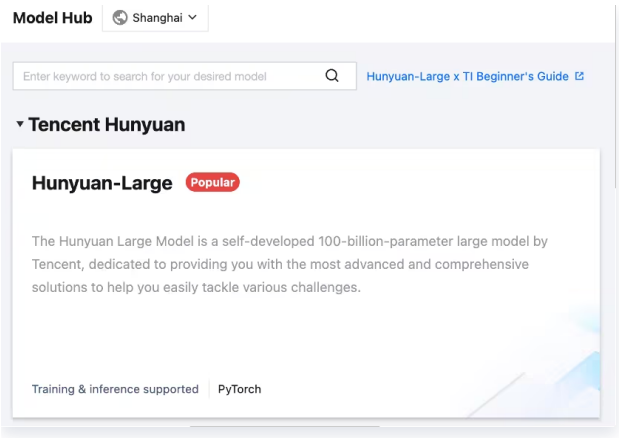
Model Hub provides the following main features:

Viewing Built-in LLMs

On the Model Hub page, the list of available models is presented in a card-based layout.

- Model tags: Models are categorized by dimensions such as task type (such as text classification, translation, or Q&A), language (such as Chinese or English), and framework (such as PyTorch). This classification is displayed in the form of tags, making it convenient for quick categorization and search.
- Model search: The platform supports a keyword-based search feature that allows you to perform fuzzy searches by model name.

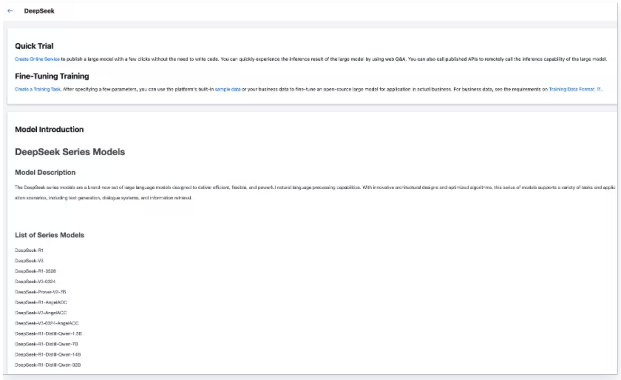
©2013-2026 Tencent Cloud. All rights reserved.



Initiating Deployment and Fine-Tuning

Click a card in Model Hub to go to the model details page, which displays more information about the model and supports features such as quick tryout and fine-tuning training.

- Model Introduction: This section includes information such as model description, a list of series models, and model ownership.
- Quick Tryout: Redirect to the **Online Services** module and deploy the model with a single click.
- Fine-Tuning Training: Redirect to the **Task-based Modeling** module and initiate fine-tuning training with a single click. Model fine-tuning significantly enhances model performance in specific scenarios and tasks, thereby meeting production-level requirements.



Model Hub > Model Details Page

Best Practices

For built-in LLMs, TI-ONE provides a wealth of best practice documentation to help you quickly get started with model deployment and fine-tuning.

- Tencent Hunyuan series:
  - Beginner's Guide on Hunyuan-Large x TI-ONE

©2013-2026 Tencent Cloud. All rights reserved.

Data Center
Data Center Introduction

Last updated: 2026-02-11 18:22:12

Data Center, as the foundational data service module of the machine learning platform, provides the following main features:

Dataset Management

- This module provides unified management of all datasets under the current root account, with the dataset list isolated by region.
- The types of modeling tasks that datasets support include:
  - Datasets for large language modeling: LLM datasets (jsonl) and MLLM datasets (jsonl)
  - Datasets for traditional CV modeling: images (jpg, png, jpeg, and bmp)
  - Datasets for large dataset modeling: table data (xls, xlsx, and csv)
  - Others: files in any format
- This module supports the preview of data content details and analysis of text data, including sample word frequency analysis and sample length analysis.

Data Labeling

- Provides a data labeling workbench for modeling of LLM and MLLM, enabling the use of schemas to flexibly define the components of the labeling workbench.
- Provides a built-in Image Classification labeling scenario, which identifies the label category to which each image in a dataset belongs, such as classifying images of cats and dogs.
- Provides a built-in Object Detection labeling scenario, which identifies the location and category of specified objects in each image within a dataset, such as drawing bounding boxes to detect and annotate cats and dogs.
- Provides a built-in Object Tracking labeling scenario, which tracks and marks a specific object across a sequence of consecutive images within a dataset.
- Provides a built-in Image Segmentation labeling scenario, which identifies the region occupied by an object belonging to a specified labeling category within each image of a dataset.
- Provides a built-in OCR labeling scenario, which supports recognition/detection of text content in images or intelligent structured labeling.

Data Construction

- Provides best practice documentation to guide you in data preprocessing.

©2013-2026 Tencent Cloud. All rights reserved.

- Provides multiple built-in algorithmic tasks, each featuring high-quality prompts and built-in self-developed data.
- Provides four types of built-in data preprocessing pipelines: supervised single-turn Q&A pipeline, supervised multi-turn Q&A pipeline, and unsupervised pipeline.
- Provides multiple data processing features, such as data cleaning, prompt optimization, data filtering, and data augmentation.

©2013-2026 Tencent Cloud. All rights reserved.

Creating a Dataset

Last updated: 2026-02-11 18:23:03

Overview

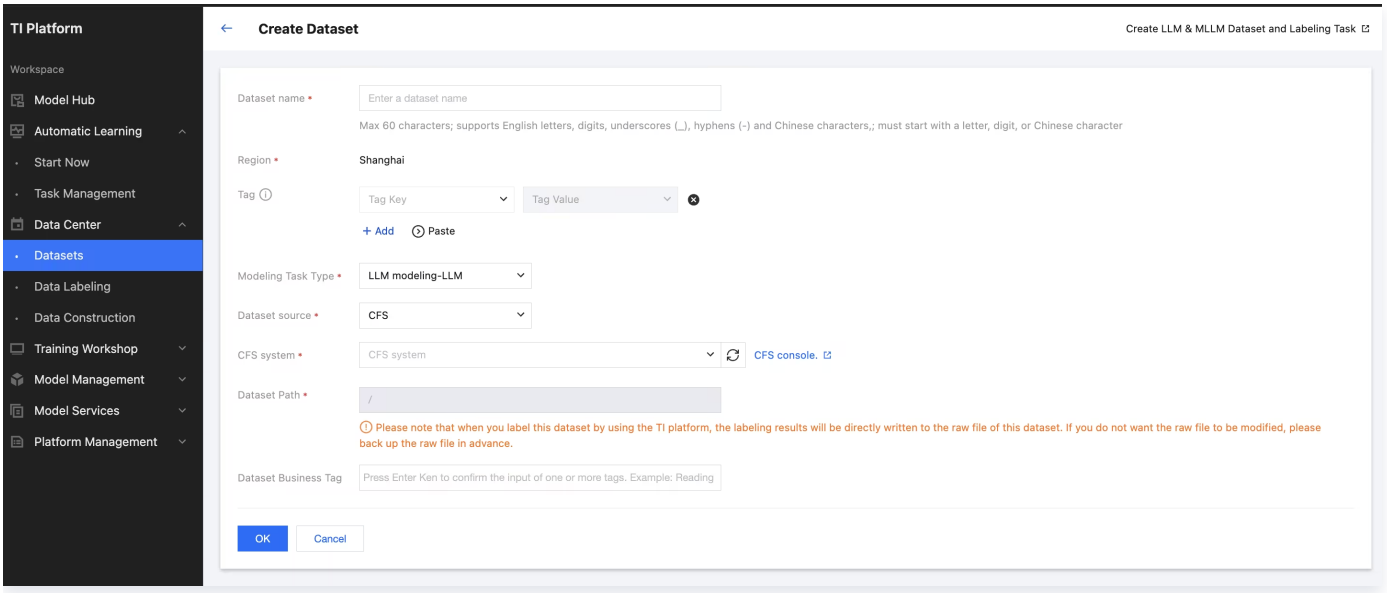
Data Center supports the following sources for creating/managing user datasets: LLM data files on Tencent Cloud File Storage (CFS) and image/table/other data files on Tencent Cloud Object Storage (COS). Therefore, the entire Data Center module employs two distinct product logics to separately support datasets for both large language modeling and traditional modeling.

> **Note**
>
> 1. COS is an object storage product with separate billing. For details, see Billing Overview in the COS documentation.
> 2. CFS is a file storage product with separate billing. For details, see Price Overview in the CFS documentation.
> 3. A public cloud account can maintain up to 200 datasets within the Data Center simultaneously in the same region. If you have expansion needs for your business, you can submit a ticket to contact us.

Datasets for Large Language Modeling

Datasets for large language modeling (LLM&MLLM) can be created by setting Modeling Task Type to Large Language Modeling. You only needs to provide the CFS path of a data file. Data Center merely associates the dataset with your CFS path without copying or dumping the original data file. When you label this LLM dataset on Tencent Cloud TI-ONE Platform (TI-ONE), label results will be written directly and in real time into the original file of your dataset. Therefore, if you do not want the original files to be modified, back up the original files in advance.

©2013-2026 Tencent Cloud. All rights reserved.



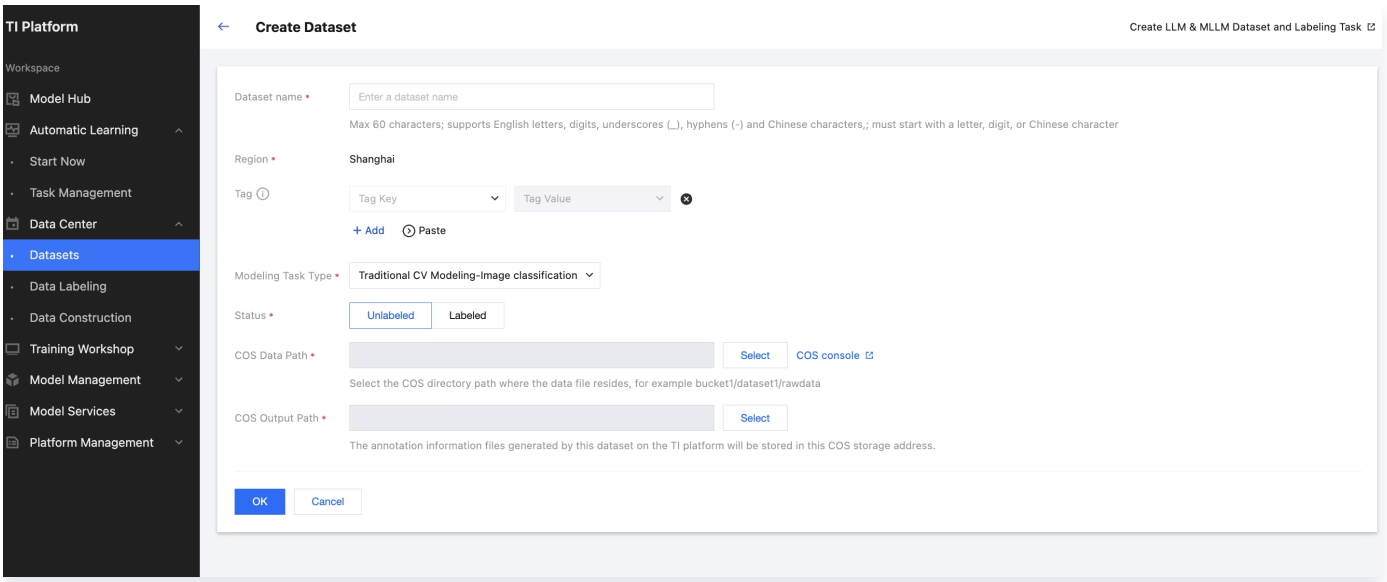
- Dataset Name: Enter a custom dataset name.
- Modeling Task Type: If Large Language Modeling – LLM is selected, this parameter serves only as a distinguishing marker for the task type and will not introduce strict differences in operation logic in subsequent processes of the product.
- Dataset CFS Path: Locate the CFS path where your data file resides. This path does not need to be precise down to the jsonl file. It only needs to accurately point to the directory level where the jsonl file is located.
- Dataset Tag: To facilitate better classification and management of large datasets, you can flexibly and freely assign tags to the datasets. Data Center will support dataset search based on these tags in the future.

Datasets for Traditional CV Modeling

Creating an Unlabeled Image Dataset

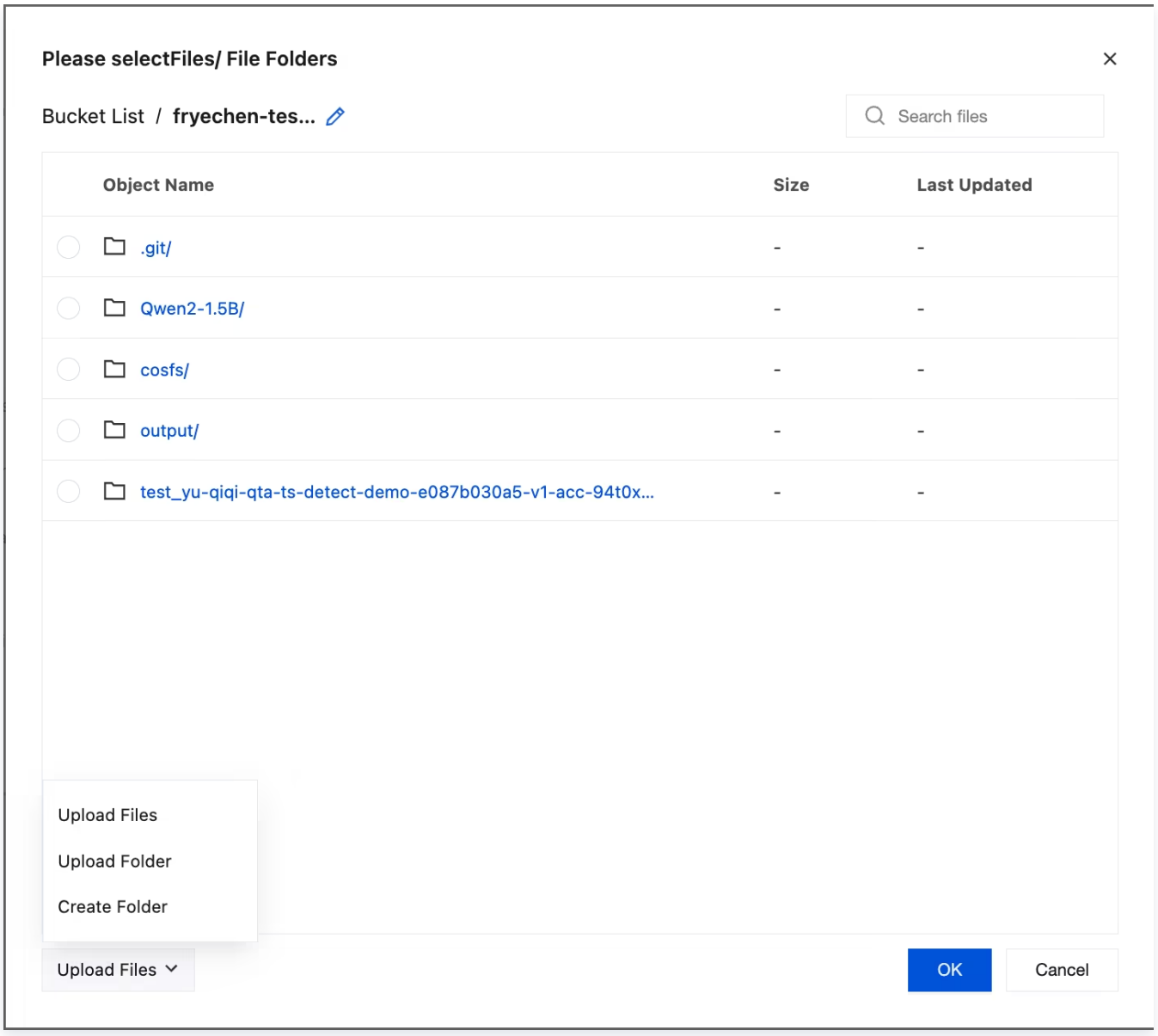
If your image data lacks relevant label information, configure a dataset as follows when creating a dataset:

©2013–2026 Tencent Cloud. All rights reserved.



- Dataset Name: Enter a custom dataset name.
- Region: The region of the dataset is the region of your current console on the frontend by default.
- Tag: This parameter reuses the Tencent Cloud Tag product, allowing you to assign different tag values to the dataset.
- Modeling Task Type: Select Traditional CV Modeling, and then select the appropriate option based on your image scenario: Image Classification, Image Detection, Image Segmentation, Image Object Tracking, or OCR.
- Status: Select Unlabeled.
- COS Data Path: Simply specify the COS path where the image data is located. If your image files are still local and have not been uploaded to the COS path, click **Select File**, select a bucket, and then select the Upload File option in the lower-left corner of the dialog box to complete the upload of local data.

©2013-2026 Tencent Cloud. All rights reserved.

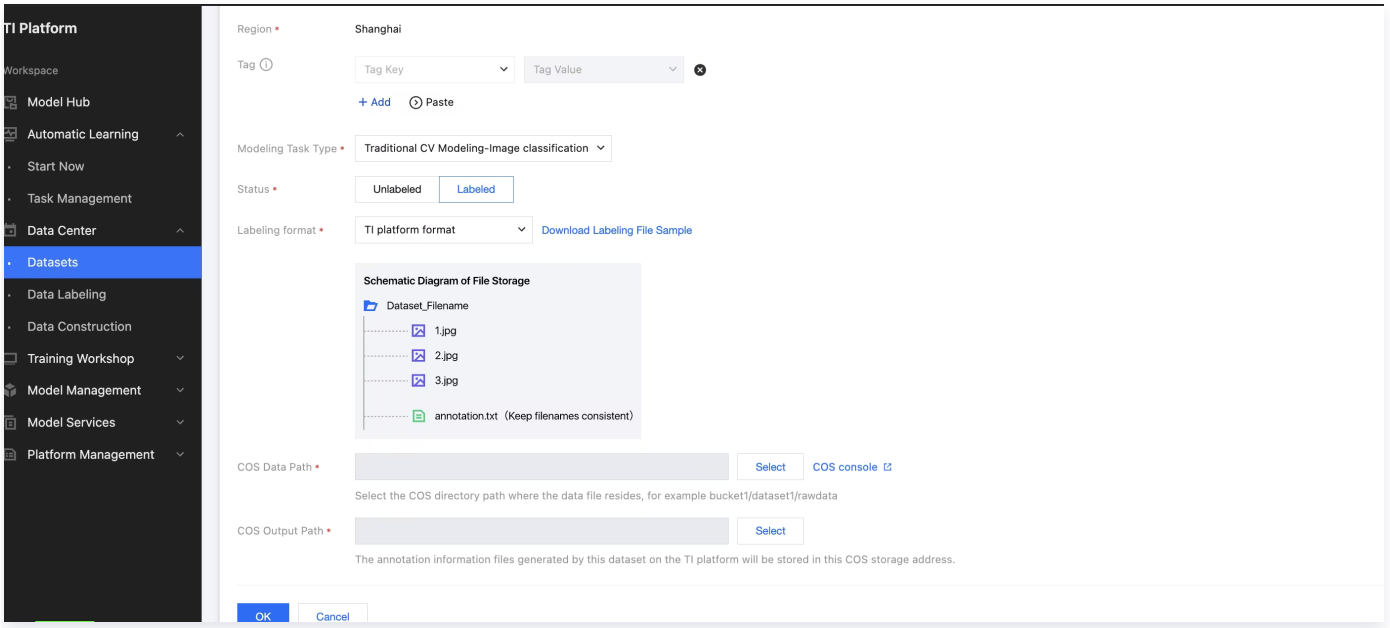


- COS Output Path: Label information that may be subsequently generated on TI-ONE for this dataset (such as when a data labeling task is created using this dataset), as well as files such as data information/snapshot records for each version of the dataset, will be stored at this COS storage address. Note: This address cannot be the same as the value of COS Path for Data Files.
- After the dataset is successfully imported, TI-ONE supports viewing dataset details on the frontend page to preview image content.

Creating an Labeled Image Dataset

If your image data contains relevant label information, configure a dataset as follows when importing the dataset:

©2013-2026 Tencent Cloud. All rights reserved.

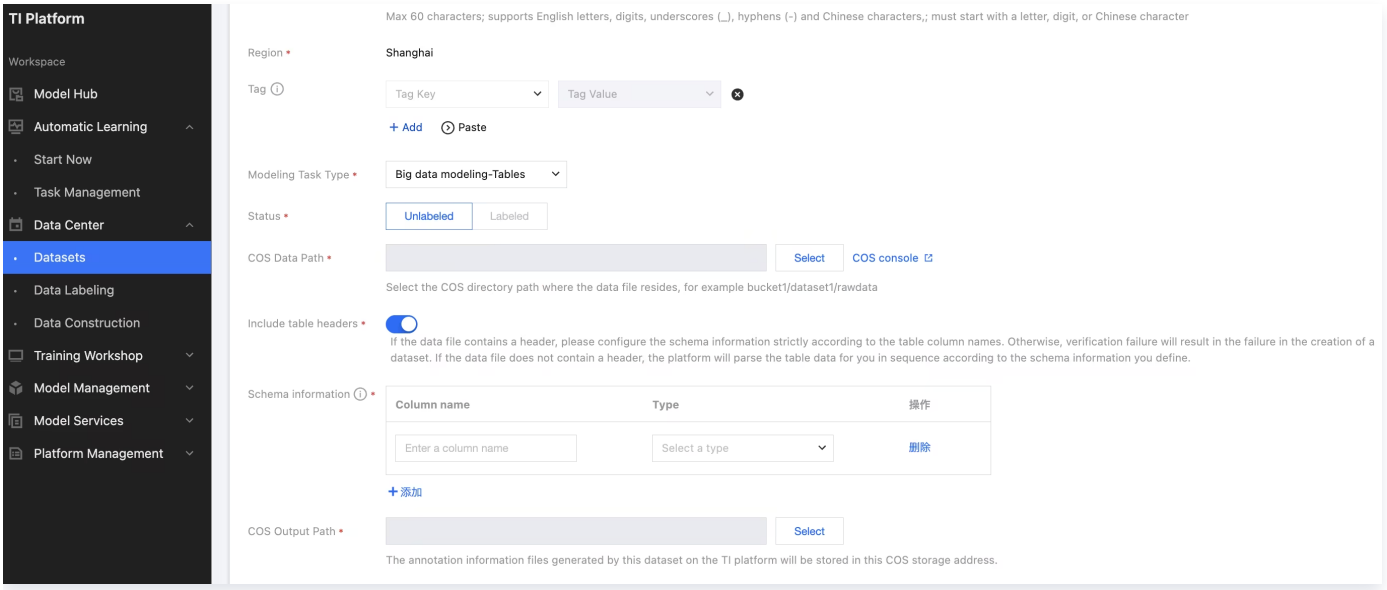


- Dataset Name: Enter a custom dataset name.
- Region: The region of the dataset is the region of your current console on the frontend by default.
- Tag: This parameter reuses the Tencent Cloud Tag product, allowing you to assign different tag values to the dataset.
- Modeling Task Type: Select Traditional CV Modeling, and then select the appropriate option based on your image scenario: Image Classification, Image Detection, Image Segmentation, Image Object Tracking, or OCR.
- Status: Select Labeled.
- Lsbeling Format: Different image scenarios require different labeling formats:
  - Depending on the labeling format, prepare your file directory according to the instructions in the corresponding diagram for file storage.
  - Click **Download Labeling File Sample** to view the specific content of the labeling file format.
- COS Data Path: Simply specify the COS path where the image data is located.
- COS Output Path: Files such as labeling information that may be subsequently generated on TI-ONE for this dataset (such as when a data labeling task is created using this dataset) will be stored at this COS storage address. Note: This address cannot be the same as the value of COS Path for Data Files.
- After the dataset is successfully imported, TI-ONE supports viewing dataset details on the frontend page to preview image content and simultaneously display a visual distribution chart of the dataset labeling information.

Datasets for Big Data Modeling

©2013-2026 Tencent Cloud. All rights reserved.

If you want to import a dataset of the Big Data Modeling - Tabular type, configure the dataset as follows:

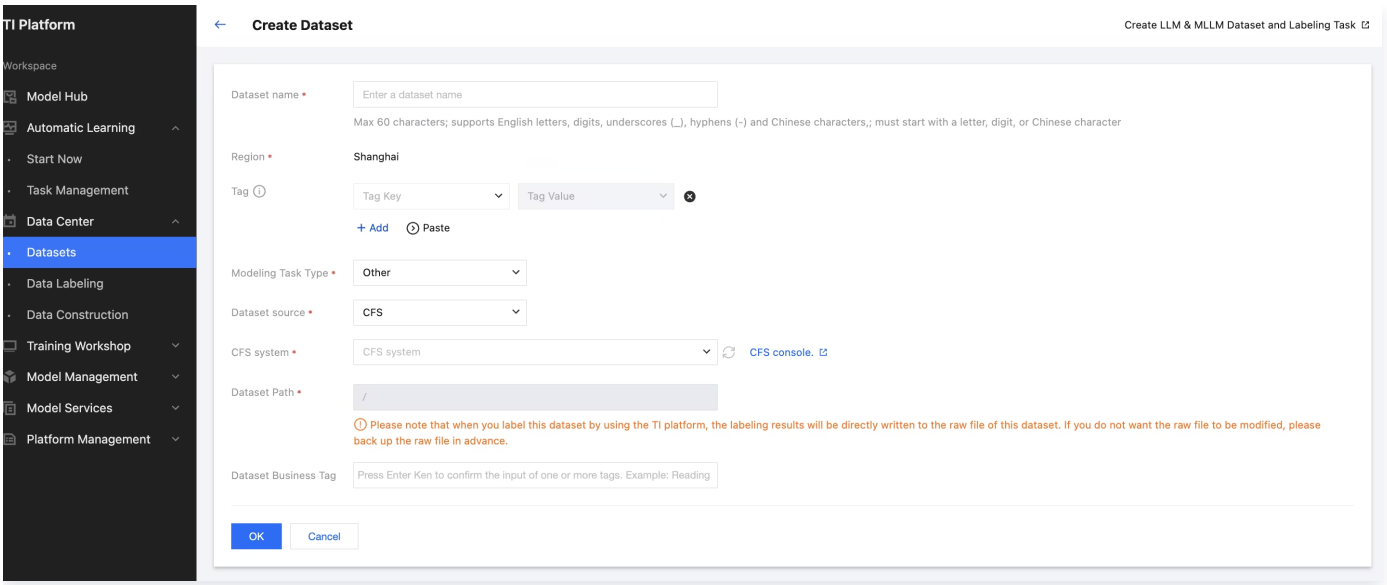


- Dataset Name: Enter a custom dataset name.
- Region: The region of the dataset is the region of your current console on the frontend by default.
- Tag: This parameter reuses the Tencent Cloud Tag product, allowing you to assign different tag values to the dataset.
- Modeling Task Type: Select Big Data Modeling - Tabular.
- Status: Select Unlabeled. Table datasets do not support uploading data files that are labeled.
- COS Data Path: Simply specify the COS path where the table data is located.
  - If the data file contains a header, enable the Header Included option and configure the schema strictly according to the table column names. Otherwise, the verification will fail, resulting in a dataset import failure. If the data file does not contain a header, disable the Header Included option, and the platform will parse the table data sequentially based on the custom schema information you specify.
- COS Output Path: Some metadata record files generated by this dataset on TI-ONE will be stored under this COS storage address. Note: This address cannot be the same as the value of COS Path for Data Files.
- After the dataset is successfully imported, TI-ONE supports viewing dataset details on the frontend page to preview the content of the table data. Additionally, TI-ONE supports displaying distribution statistics for enumerable columns.

Other Datasets

If your file extension does not meet the platform's built-in data format requirements, or if you do not require TI-ONE to display data details and only need to manage a data path, set Modeling Task Type to Other when creating a dataset. Configure the dataset as follows:

©2013-2026 Tencent Cloud. All rights reserved.



- Dataset Name: Enter a custom dataset name.
- Region: The region of the dataset is the region of your current console on the frontend by default.
- Tag: This parameter reuses the Tencent Cloud Tag product, allowing you to assign different tag values to the dataset.
- Modeling Task Type: Select Other.
- Status: Only Unlabeled can be selected.
- COS Path for Data Files: Simply specify the COS path where your data file is located.
- COS Output Path: Some metadata record files generated by this dataset on TI-ONE will be stored at this COS storage address. Note: This address cannot be the same as the value of COS Path for Data Files.
- After the dataset is successfully imported, TI-ONE does not support viewing dataset details on the frontend page.

©2013-2026 Tencent Cloud. All rights reserved.

Dataset List Operations

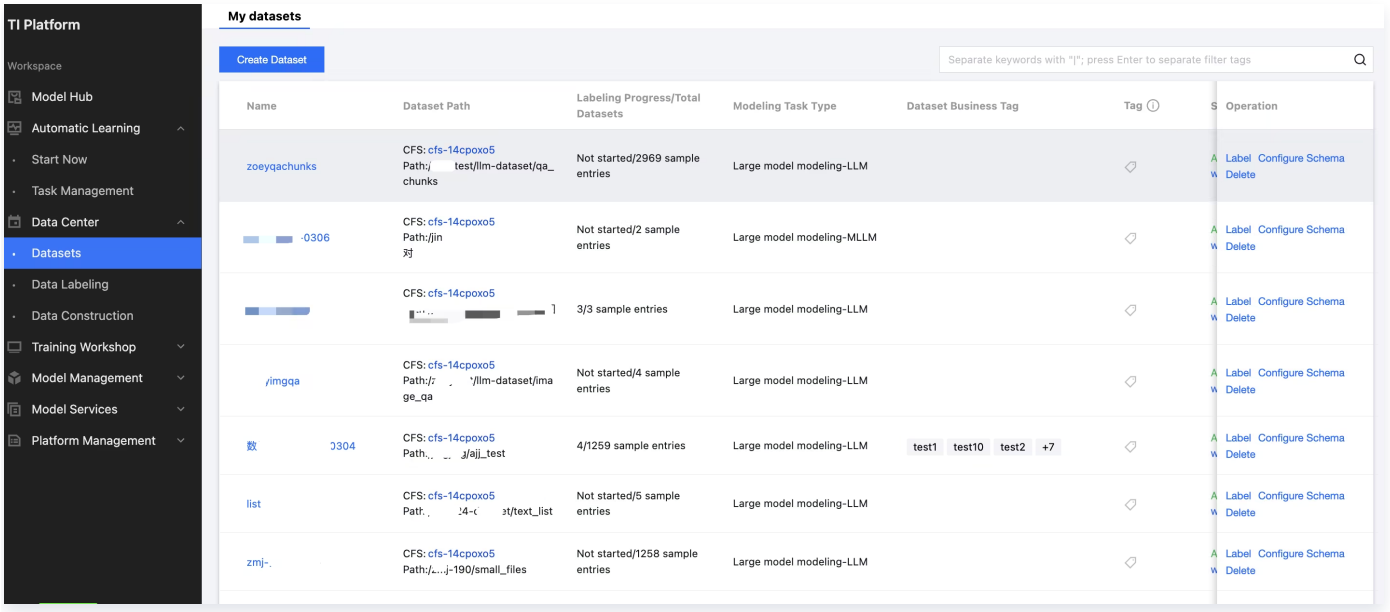
Last updated: 2026-02-11 18:23:29

Overview

The dataset list page provides unified management of the collection of all datasets under the user account, displays key dataset information, supports operations such as label delete, and configuring schemas, and allows you to click the name of a dataset to go to the dataset details page.

List Fields

The fields displayed on the dataset list page and their meanings are explained as follows:



- Name: displays the custom name of a dataset, and supports clicking the name to go to the dataset details page.
- Annotation Progress/Total Dataset Size:
  - Datasets for large language modeling: According to the schema parsing rules, a complete sample is counted as 1. Samples labeled in the labeling workbench are counted as "labeled".
  - Dataset for Traditional CV Modeling (Images): counts the number of all images under the specified path. Images with labeling results are counted as "labeled".
  - Dataset for Big Data Modeling (Table): counts the number of rows.
  - Other: counts the number of all files under the specified path.
- Modeling Task Type: a custom parameter for creating a dataset.
- Dataset Tag:

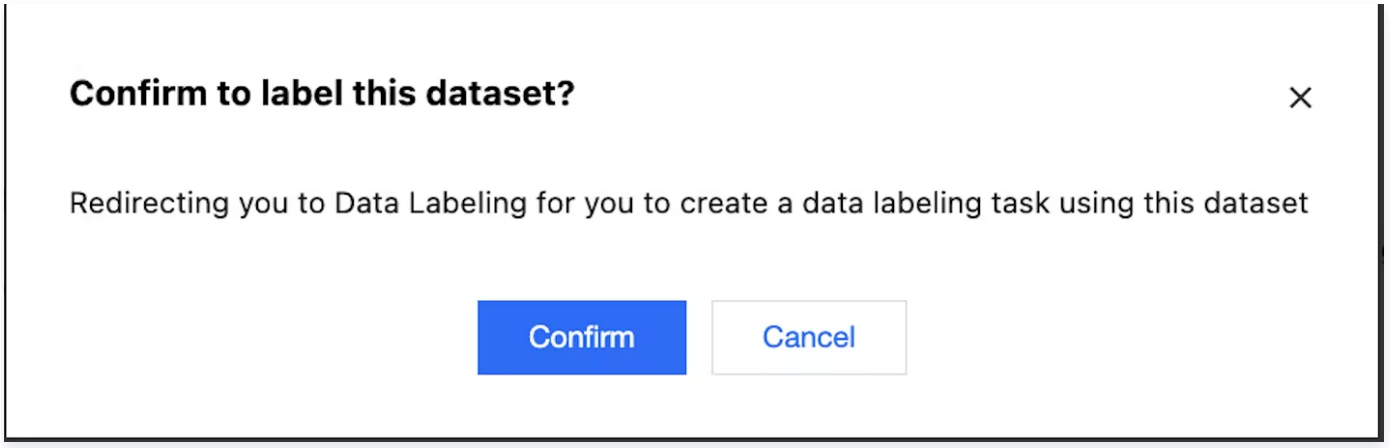
©2013-2026 Tencent Cloud. All rights reserved.

  - Dataset for Large Language Modeling: custom tag content defined on the configuration page for creating a dataset.
  - Dataset for Traditional CV Modeling: By default, tags are added according to the image scenario to which the dataset belongs. Enumeration values include "image classification, image detection, image segmentation, image object tracking, and OCR".
- Tag: displays the Tencent Cloud CAM tag selected during dataset import.
- Status:
  - Datasets for Large Language Modeling
    - Upon successful creation, the dataset enters the Available state. However, you need to complete the schema configuration operation to view dataset details and perform labeling operations.
  - Datasets for Traditional CV Modeling
    - Import XX%: displays the progress of dataset import in real time in percentage form, from when you click **Confirm** on the import page until the dataset is successfully imported.
    - Available: The dataset transitions to the Available status once it is imported successfully or after the data source is successfully synchronized. It remains in the Available status if the label operation fails, the download operation fails, or the new version release operation fails. If the data source synchronization operation fails, the dataset transitions to the Failed status.
    - Failed: supports displaying the detailed reason for import failure or data source synchronization failure in a floating window.
    - Unavailable: The dataset is in the Unavailable status when it is being deleted.
- Creation Time: records the time when the dataset was created and supports user-selected sorting in ascending or descending order for display.
- Operations: The following provides a detailed introduction to the operation feature.

Operations - Annotation

This operation allows you to create data labeling tasks with a single click using this dataset.

©2013-2026 Tencent Cloud. All rights reserved.



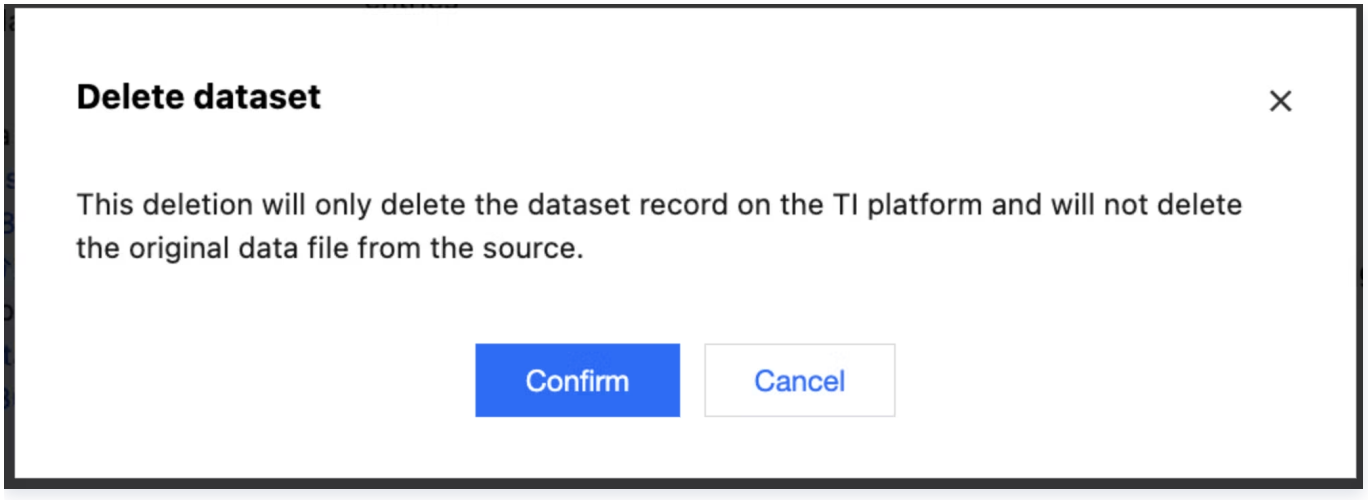
- If it is a dataset for large language modeling, after you click **Confirm**, the platform will automatically create the corresponding labeling task and redirect to the labeling workbench upon successful creation.
- If it is a dataset for traditional CV modeling, after **Confirm** is clicked, the platform will redirect to the Data Center > Data Annotation > Create Annotation Task configuration page from within the current page. The dataset will be selected by default and cannot be modified.

Note

- Only one labeling task can be created for a dataset at a time.
- Only users with write permissions on the corresponding Cloud Object Storage (COS) or Cloud File Storage (CFS) path of this dataset can create data labeling tasks using this dataset.

Operations – Deletion

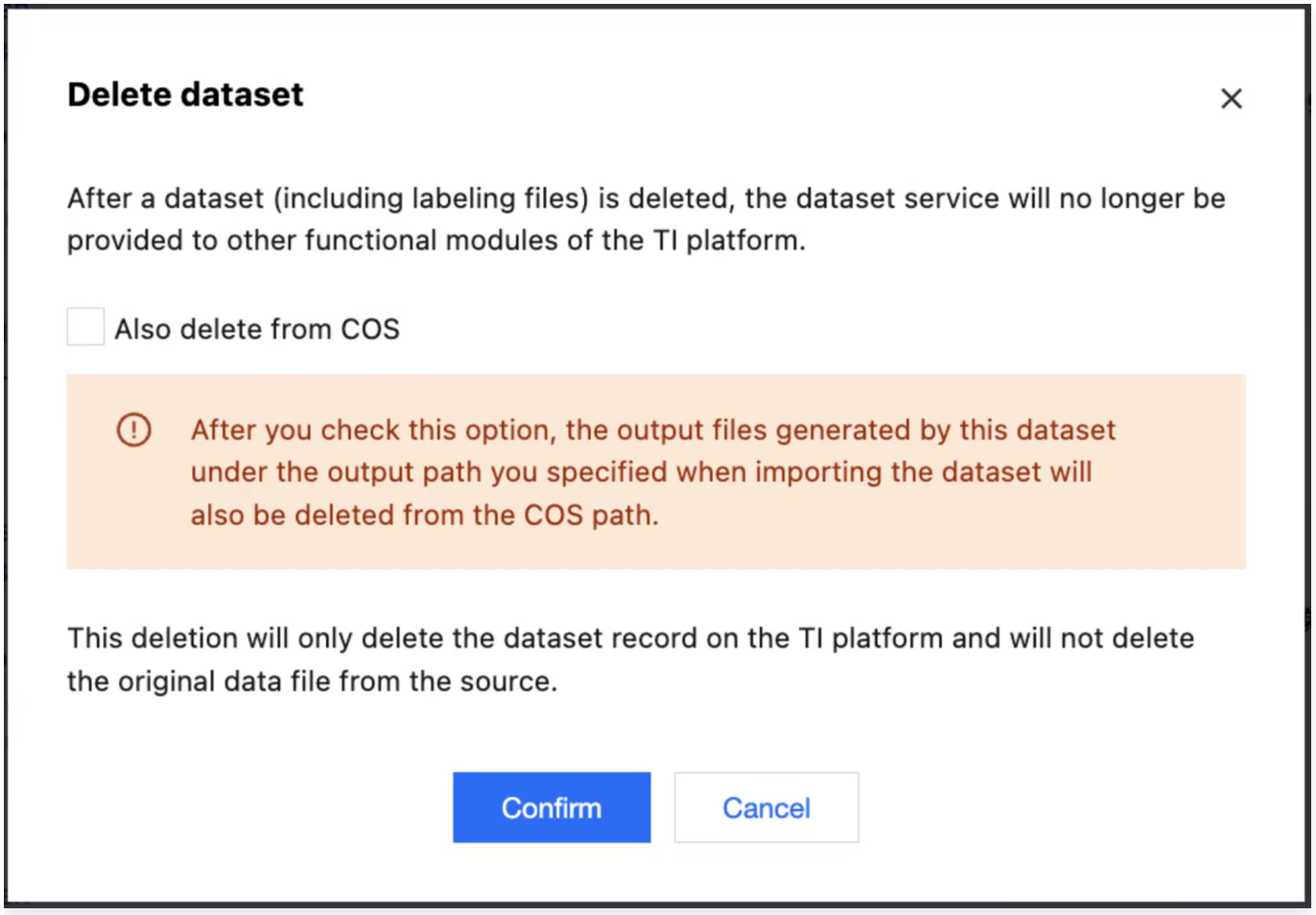
If it is an LLM dataset: Deleting this dataset will not affect the original data files stored on CFS, but only deletes this dataset record from Tencent Cloud TI-ONE Platform (TI-ONE).



If it is a non-LLM dataset:

©2013-2026 Tencent Cloud. All rights reserved.

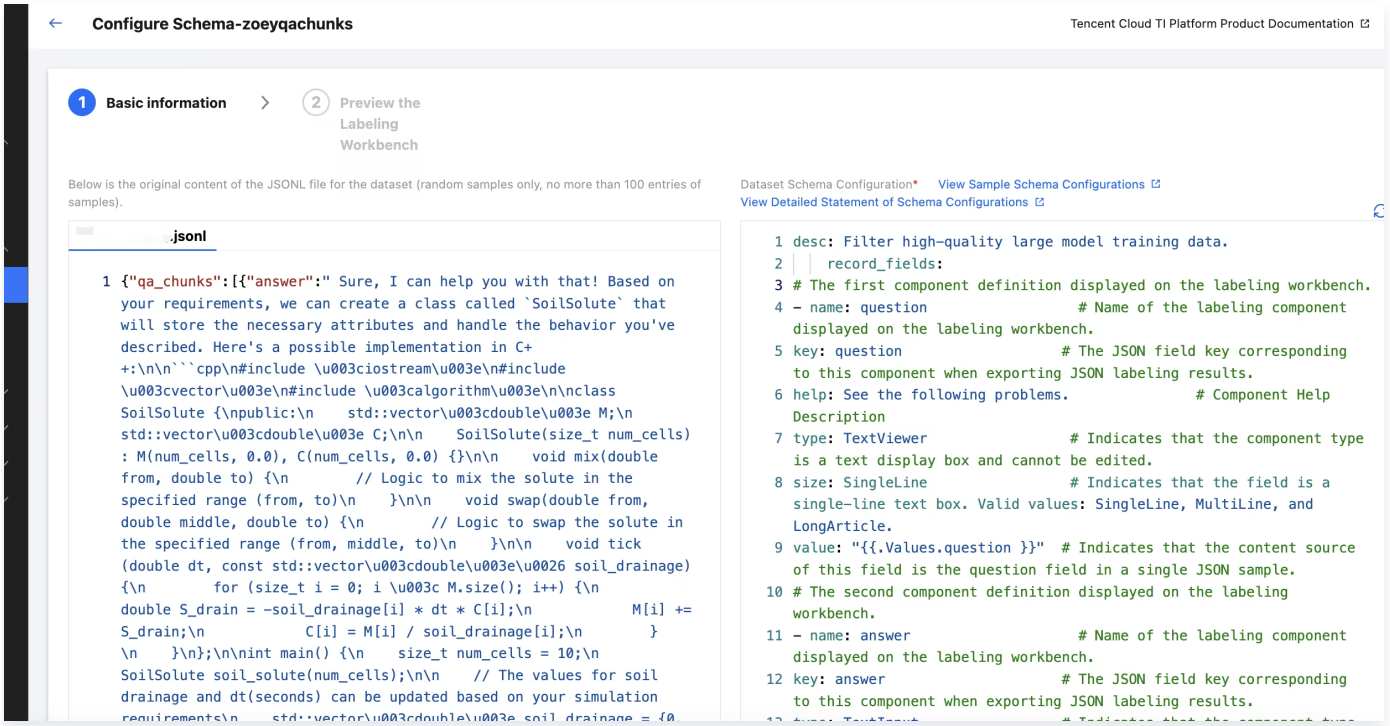
- The background unbinds the corresponding COS path from the dataset on TI-ONE and deletes this dataset record.
- (Optional) When the dataset is deleted, the background automatically cleans the files in the COS bucket under the output path defined by the dataset. Only files under the output path are automatically cleaned, without affecting files in the original input path.



Operations – Schema Configuration

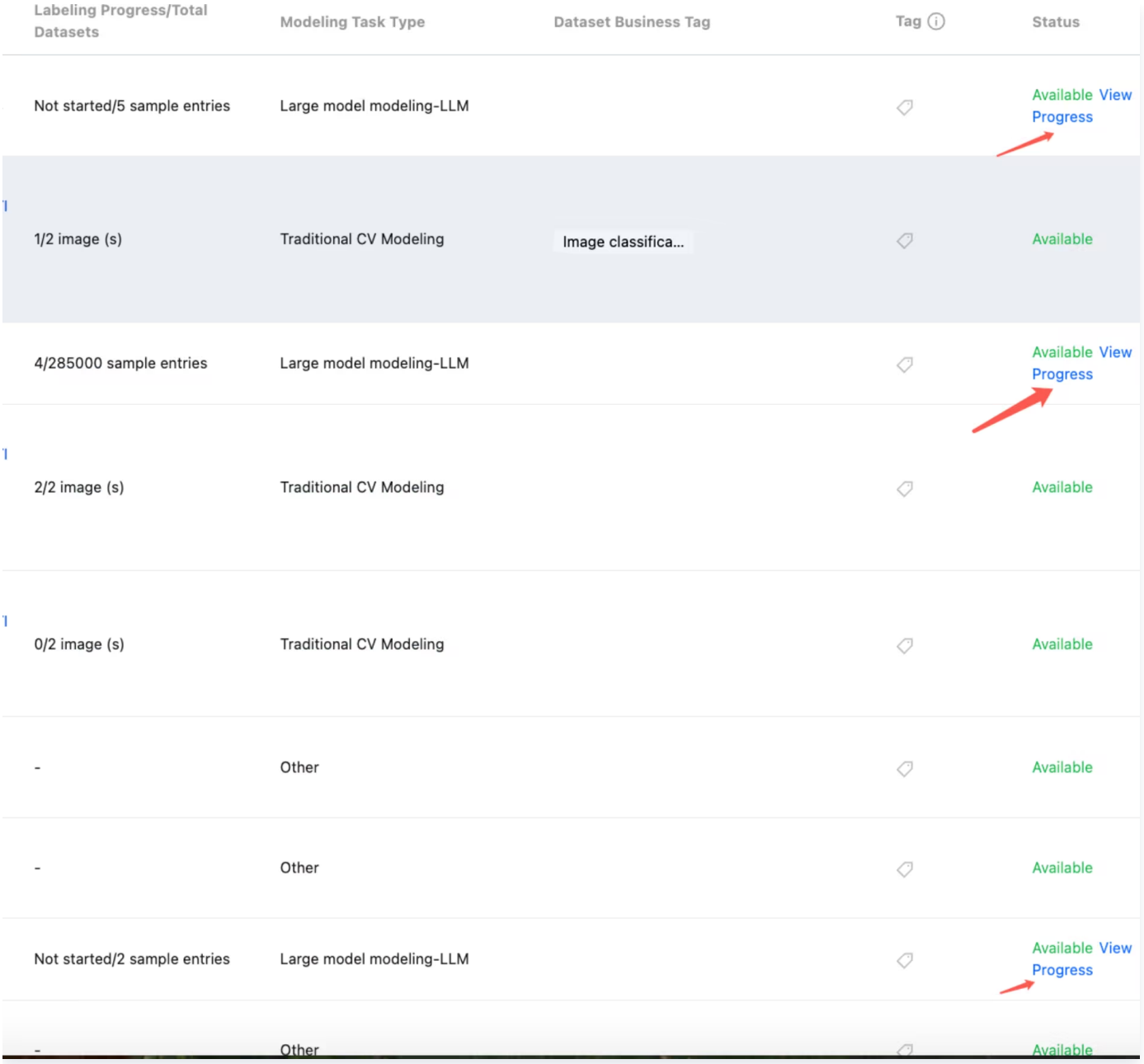
This operation is available only for datasets for large language modeling, and allows you to define various complex LLM&MLLM data content by customizing the schema information of the dataset. For schema configuration rules, see Detailed Syntax of Schema Configuration. On the left side of the configuration page, the platform displays partial content of your original data to assist you in referencing the left side while rewriting the schema configuration on the right side. Additionally, you can click **Next: Preview Labeling Workbench** to view the configuration effect of the schema in real time, ensuring that the configuration meets expectations. If you find that the labeling workbench does not meet expectations, you can click to go back to **Previous: Basic Information** to modify the configuration.

©2013-2026 Tencent Cloud. All rights reserved.



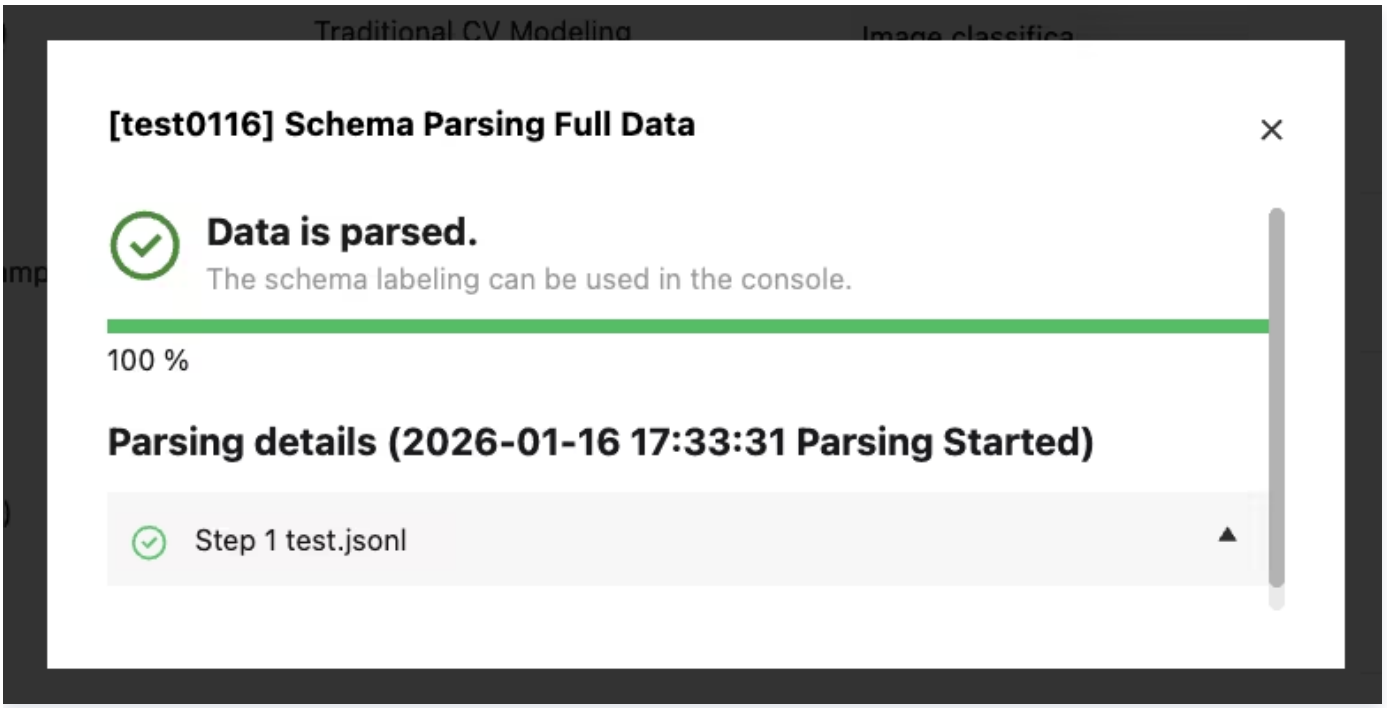
After the schema of the dataset is successfully configured, the page will automatically return to the dataset list. At this point, the platform will parse all samples in the data files based on your configured schema information. You can click **View Progress** in the Status column on the list page to view the progress of full parsing in real time.

©2013-2026 Tencent Cloud. All rights reserved.



Labeling Progress/Total Datasets	Modeling Task Type	Dataset Business Tag	Tag	Status
Not started/5 sample entries	Large model modeling-LLM			Available View Progress
1/2 image (s)	Traditional CV Modeling	Image classifica...		Available
4/285000 sample entries	Large model modeling-LLM			Available View Progress
2/2 image (s)	Traditional CV Modeling			Available
0/2 image (s)	Traditional CV Modeling			Available
-	Other			Available
-	Other			Available
Not started/2 sample entries	Large model modeling-LLM			Available View Progress
-	Other			Available

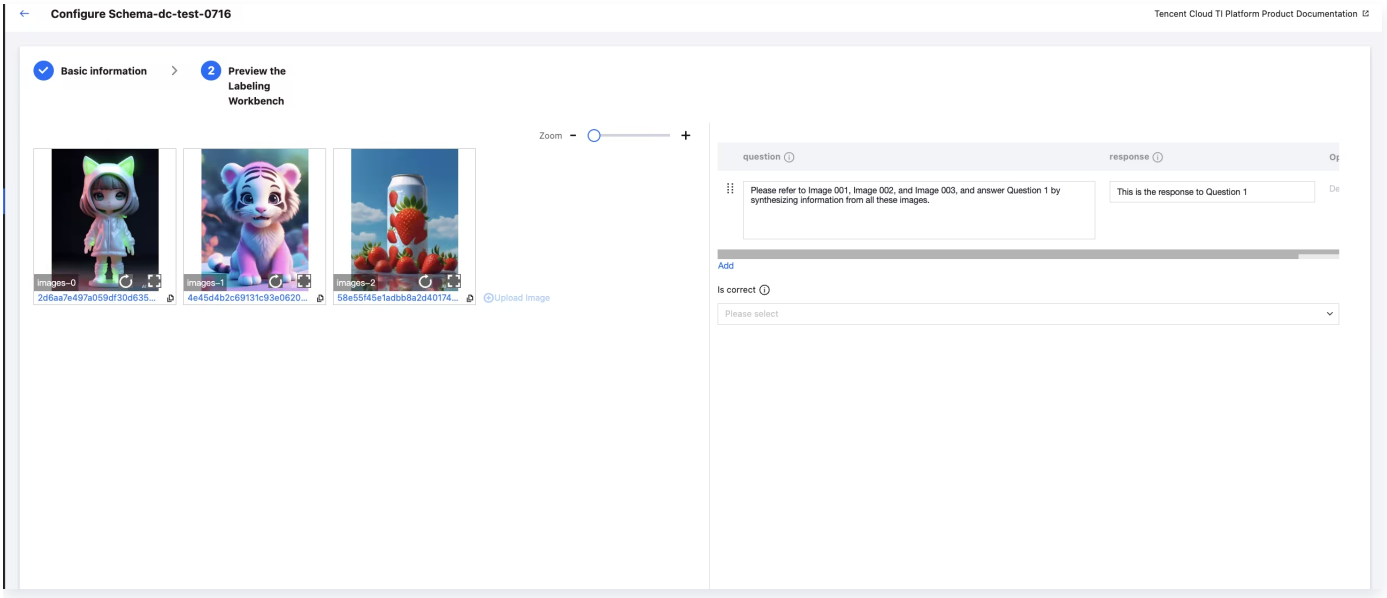
©2013-2026 Tencent Cloud. All rights reserved.



Dataset Details Page

Datasets for Large Language Modeling

Clicking the name of a dataset for large language modeling allows you to view the dataset details, and the displayed content is presented according to the configured schema. For multimodal datasets, images and corresponding text content are directly paired and displayed in the same row to enhance the readability of data samples.



Datasets for Traditional CV Modeling

©2013-2026 Tencent Cloud. All rights reserved.

> **Note**
> For the Detail Preview and Data Pivot features on the dataset details page, COS provides 10T of free detail preview traffic per account each month. Any usage beyond this limit will incur charges. For details, see COS Billing Rules.

Clicking the name of the image dataset allows you to view the dataset details. The details page contains three main sections:

- Basic Information: This section displays the key information of the dataset.
- Visualization of Annotation Information: If the current dataset is bound with labeling information for Image Classification/Object Detection/Image Segmentation, this module will be displayed. In other scenarios, this module is automatically hidden. Note: The maximum number of tag values that can be statistically tracked in the background is 20 (displayed as Top 20 by proportion). Any categories beyond 20 will be grouped into the Other category.
- Details Display: This section allows you to preview a list of the first 2000 images in the dataset. It also supports filtering by labeling status and by specified labeling categories for

©2013-2026 Tencent Cloud. All rights reserved.

display.

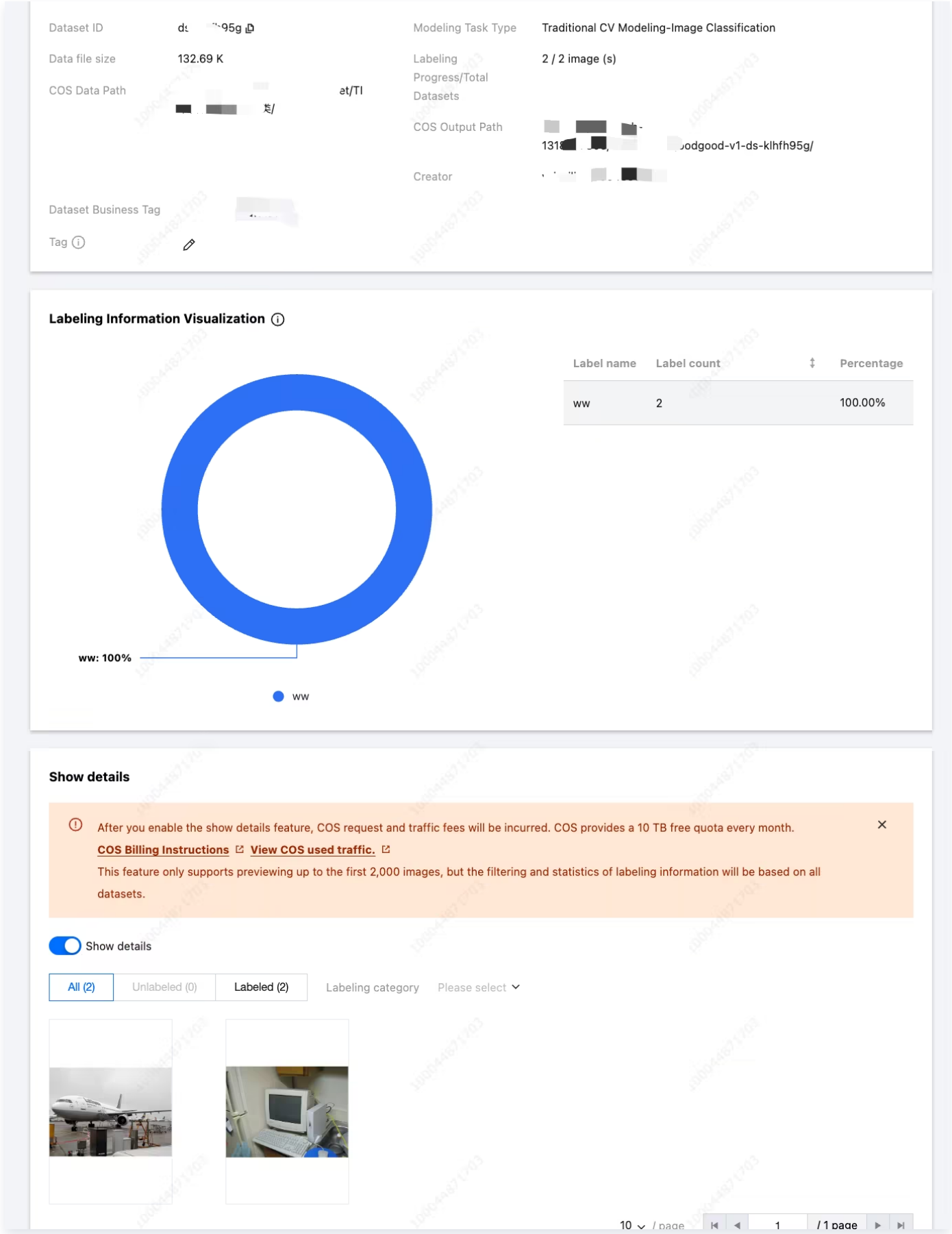
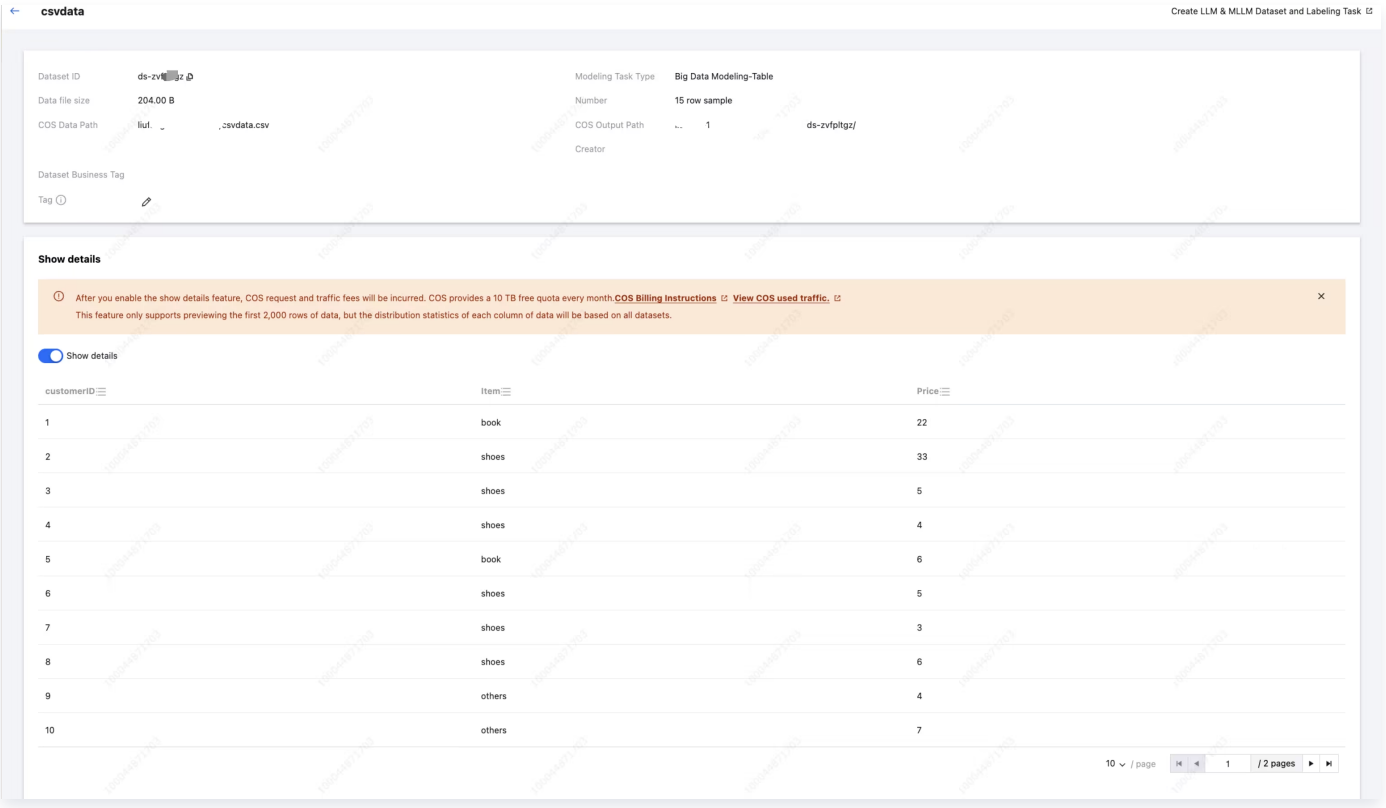

Dataset ID
Data file size
132.69 K
COS Data Path
Modeling Task Type
Traditional CV Modeling-Image Classification
Labeling Progress/Total Datasets
2 / 2 image (s)
COS Output Path
Creator
Dataset Business Tag
Tag
Labeling Information Visualization
Label name
Label count
Percentage
ww
2
100.00%
ww: 100%
ww
Show details
After you enable the show details feature, COS request and traffic fees will be incurred. COS provides a 10 TB free quota every month.
COS Billing Instructions
View COS used traffic.
This feature only supports previewing up to the first 2,000 images, but the filtering and statistics of labeling information will be based on all datasets.
Show details
All (2)
Unlabeled (0)
Labeled (2)
Labeling category
Please select
10 / page
1
/ 1 page

Table Datasets for Big Data

Clicking the name of a table dataset allows you to view the dataset details. The details page contains two main sections:

- Basic Information: This section displays the key information of the dataset.
- Details Display: This section allows you to preview the first 2,000 rows of table content in the dataset. For enumerable columns, it supports clicking to view column data distribution information. The statistical analysis covers the entire dataset, not just the 2,000 rows previewed on the frontend.



Other Types of Datasets

Clicking the names of other types of datasets takes you to the dataset details page, which only displays the basic information of the dataset. Since there are no restrictions on the import format for this type of dataset, the details page does not support content preview.

©2013–2026 Tencent Cloud. All rights reserved.

Creating a Data Labeling Task for an LLM

Last updated: 2026-02-11 18:23:42

Overview

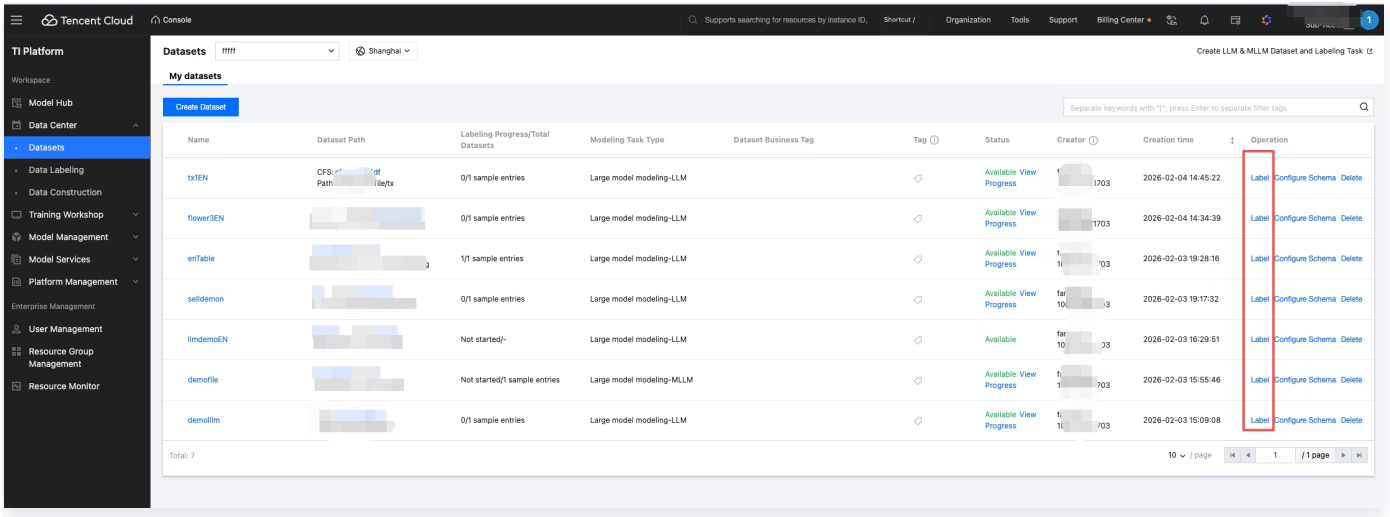
Tencent Cloud TI-ONE Platform (TI-ONE) provides the labeling feature for Large Language Model (LLM) and Multimodal Large Language Model (MLLM) datasets, allowing you to customize the schema information of a dataset to flexibly build a custom labeling workbench.

Creating an Labeling Task

Choose **Data Center > Datasets** module, select the created LLM dataset, and then choose Operation > Lebel. The backend will automatically create the corresponding labeling workbench based on the schema configuration information of the dataset.

Must-Knows:

- The Data Center only associates this dataset with your Cloud File Storage (CFS) path and does not copy or dump your original data files.
- When you label this dataset on TI-ONE, labeling results will be written directly and in real time into the original files of your dataset. Therefore, if you do not want the original files to be modified, you need to back up the original files in advance.



LLM Label Workbenchs

The flexible schema configurations of TI-ONE support labeling scenarios, including but not limited to: filtering of high-quality text Q&A pairs, text data cleansing, reviewing or modification

©2013-2026 Tencent Cloud. All rights reserved.

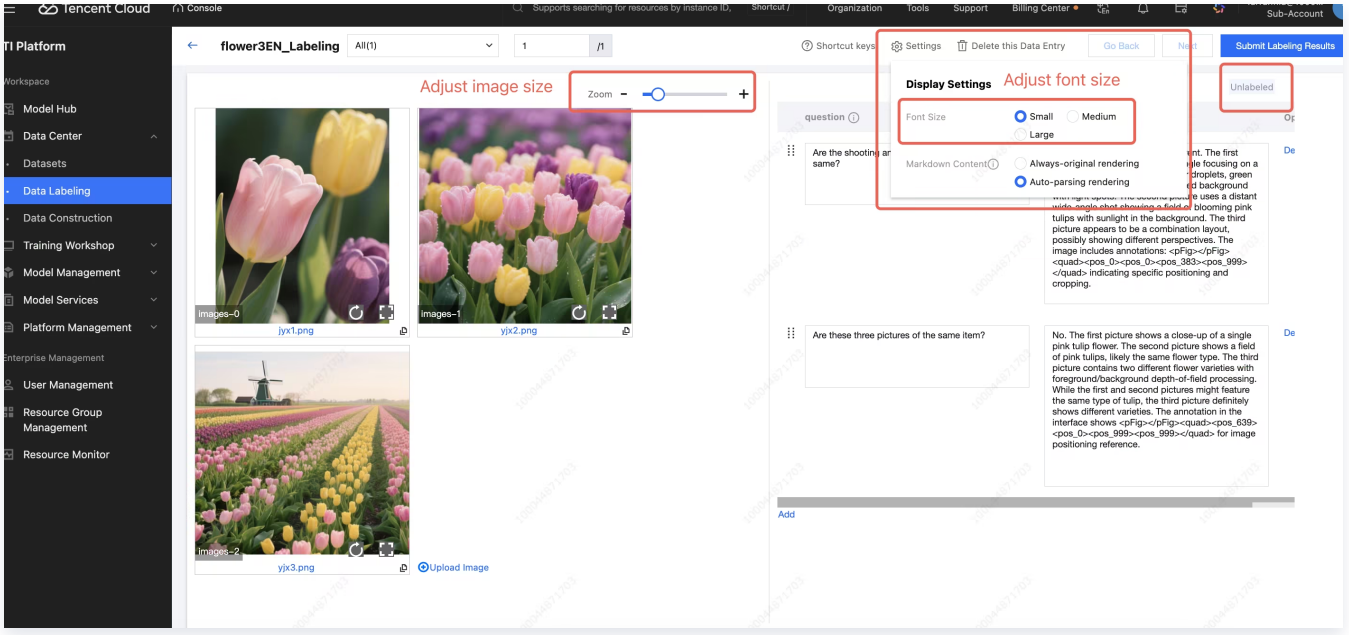
of image Q&A pairs, evaluation of image Q&A competitors, image multi-turn Q&A, and multimodal reading comprehension for image text descriptions.
The following are 3 examples of labeling benchs corresponding to different schema configurations:

1. Multi-image multi-turn Q&A

```
desc: Multi-image multi-turn Q&A
record_fields:
  - name: img  # Annotation component name displayed in the annotation
console.
    key: img  # Name of the JSON field for exporting annotation results.
    type: ImageListInput  # Component type
    help: "Please add the field description."  # Component help
description
    value:
      {{- range .Values.img }}  # Use a loop to reference the image
list.
      - {{ . }}
      {{- end }}
  - name: target  # Annotation component name displayed in the
annotation console.
    key: target  # Name of the JSON field for exporting annotation
results.
    type: List  # Component type
    help: "Please add the field description."  # Component help
description
    value:
      {{- range .Values.target }}  # Use a loop to expand the contents
of the list.
    -
      - name: Question  # Annotation component name displayed in the
annotation console.
        key: question  # Name of the JSON field for exporting annotation
results.
        type: TextInput  # Component type
        help: "Please add the field description."  # Component help
description.
        value: "{{ .question }}"
        size: MultiLine
      - name: Answer  # Annotation component name displayed in the
annotation console.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        key: answer  # Name of the JSON field for exporting annotation
results.
        type: TextInput  # Component type.
        help: "Please add the field description."  # Component help
description.
        value: "{{ .answer }}"
        size: LongArticle
      {{- end }}
```

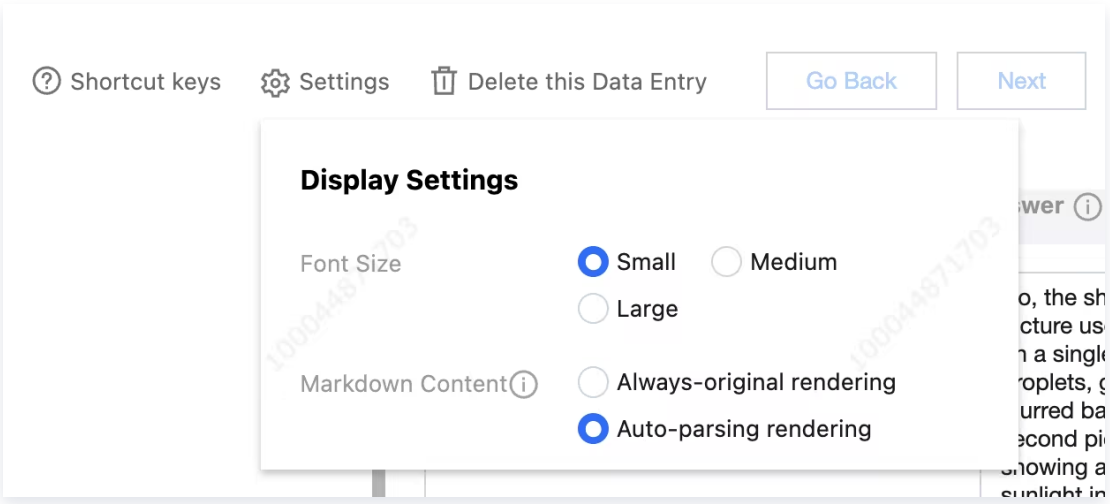


In this scenario, you can configure schemas to display multiple images and Q&A pairs, and you can add, delete, or reorder Q&A pairs. In addition, you can also configure different input box sizes based on the text length of the question and answer fields.

Detailed features of an labeling workbench:

- Click Zoom View above the image area on the left side to adjust the display size of an image.
- Click Settings on the right side of the title bar to configure the font size and whether to display the markdown content in the text.

©2013-2026 Tencent Cloud. All rights reserved.



- Click to switch between the Unlabeled and Labeled statuses. If there are modifications to the labeling content in the current sample, the backend will automatically change the status to Labeled. If there are no modifications in the current sample, you can also manually click Unlabeled to change the status to Labeled.

2. Multi-model evaluation for a single image

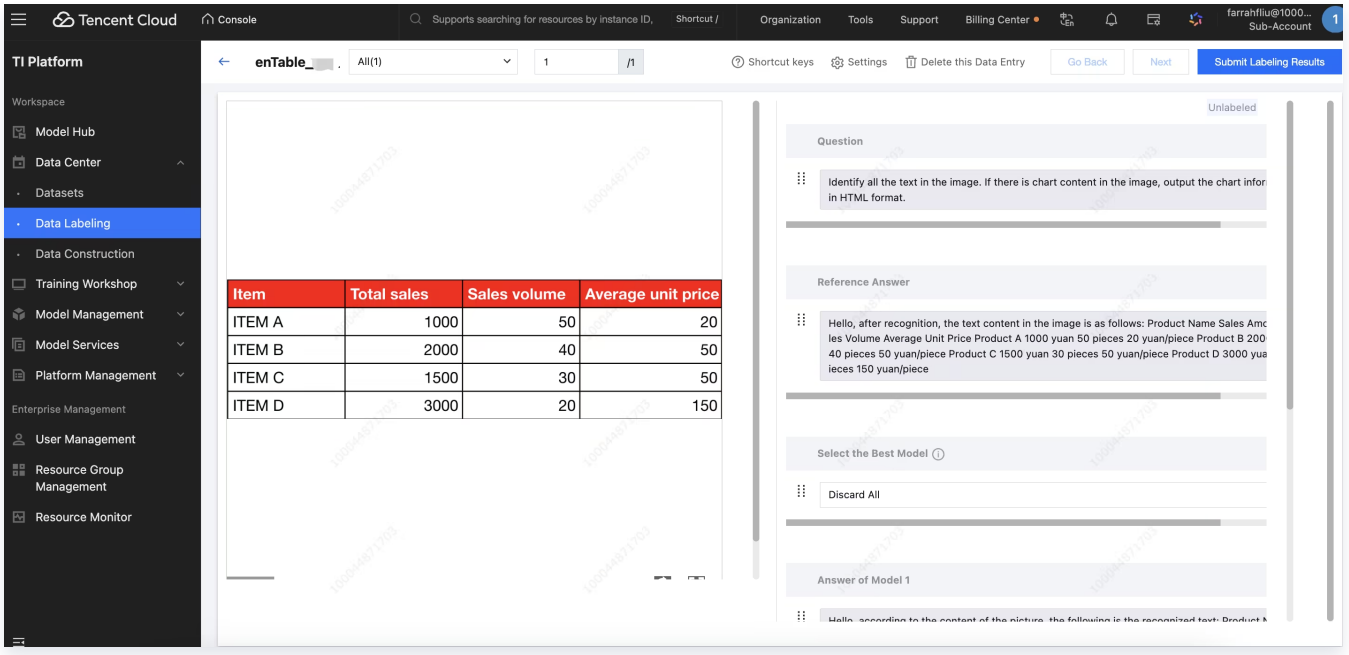
```
desc: Automatically generated YAML
record_fields:
  - name: Image  # Annotation component name displayed in the annotation
console.
    key: Images  # Name of the JSON field for exporting annotation
results. (The name can contain letters and underscores and cannot start
with a digit.)
    type: ImageViewer  # Component type.
    help: "Please add the field description."  # Component help
description
    value: "{{ .Values.Images }}"
  - name: Question  # Annotation component name displayed in the
annotation console.
    key: Query  # Name of the JSON field for exporting annotation
results. (The name can contain letters and underscores and cannot start
with a digit.)
    type: TextInput  # Component type
    help: "Please add the field description."  # Component help
description.
    value: "{{ .Values.Query }}"
    size: MultiLine
  - name: Reference answer  # Annotation component name displayed in the
```

©2013-2026 Tencent Cloud. All rights reserved.

```
annotation console.
    key: sn_vl_0_6_0_10b_8k_beta_0624  # Name of the JSON field for
exporting annotation results. (The name can contain letters and
underscores and cannot start with a digit.)
    type: TextInput  # Component type
    help: "Please add the field description."  # Component help
description.
    value: "{{ .Values.gtmodel }}"
    size: LongArticle
  - name: Select the best model.                # Annotation component
name displayed in the annotation console.
    key: correct_model             # JSON field key corresponding to the
component when JSON annotation results are exported.
    type: StringSelector            # Indicates that the component type
is a string selection component.
    option: SingleSelector           # Indicates that this component
allows multiple selections. Valid values of this field:
SingleSelector/MultiSelector
    help: Model evaluation             # Component help description.
    choices:                        # Specify the option content.
      - Model 1
      - Model 2
      - Model 3
      - Discard all
  - name: Answer of model 1  # Annotation component name displayed in
the annotation console.
    key: kzx2npurd5  # Name of the JSON field for exporting annotation
results. (The name can contain letters and underscores and cannot start
with a digit.)
    type: TextInput  # Component type
    help: "Please add the field description."  # Component help
description.
    value: '{{ index .Values.model1 }}'
    size: MultiLine
  - name: Answer of model 2  # Annotation component name displayed in
the annotation console.
    key: FT_qw15_sft_0626_v100_800  # Name of the JSON field for
exporting annotation results. (The name can contain letters and
underscores and cannot start with a digit.)
    type: TextInput  # Component type.
    help: "Please add the field description."  # Component help
```

©2013-2026 Tencent Cloud. All rights reserved.

```
description.
    value: "{{ .Values.model2 }}"
    size: MultiLine
  - name: Answer of model 3  # Annotation component name displayed in
the annotation console.
    key: V4_FT_qw15_sft_0726_temp_old_500  # Name of the JSON field for
exporting annotation results. (The name can contain letters and
underscores and cannot start with a digit.)
    type: TextInput  # Component type.
    help: "Please add the field description"  # Component help
description.
    value: "{{ .Values.model3 }}"
    size: MultiLine
```



In this scenario, you can configure schemas to display a single test image and the inference results of different models, and configure the names of the models to be evaluated.

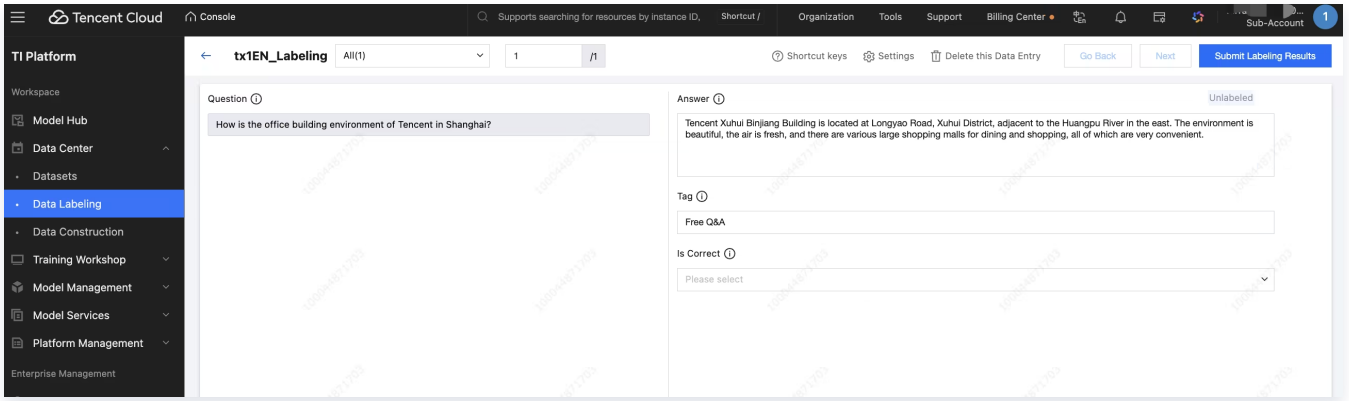
3. Filtering of high-quality text Q&A pairs

```
desc: Automatically generated YAML.
record_fields:
  - name: Question  # Annotation component name displayed in the
annotation console.
    key: question  # Name of the JSON field for exporting annotation
results. (The name can contain letters and underscores and cannot start
with a digit.)
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    type: TextViewer  # Component type.
    help: "Please add the field description."  # Component help
description.
    value: "{{ .Values.question }}"
    size: MultiLine
  - name: Answer  # Annotation component name displayed in the
annotation console.
    key: answer  # Name of the JSON field for exporting annotation
results. (The name can contain letters and underscores and cannot start
with a digit.)
    type: TextInput  # Component type
    help: "Please add the field description."  # Component help
description
    value: "{{ .Values.answer }}"
    size: MultiLine
  - name: tag  # Annotation component name displayed in the annotation
console.
    key: tag  # Name of the JSON field for exporting annotation results.
(The name can contain letters and underscores and cannot start with a
digit.)
    type: TextInput  # Component type.
    help: "Please add the field description."  # Component help
description.
    value: "{{ .Values.tag }}"
    size: SingleLine
  - name: Correct or not                    # Annotation component name
displayed in the annotation console.
    key: correct                    # JSON field key corresponding to
the component when JSON annotation results are exported.
    type: StringSelector            # Indicates that the component type
is a string selection component.
    help: Please determine whether the answer is correct.           #
Component help description.
    option: SingleSelector          # Indicates that this component
allows only single selection. Valid values of this field:
SingleSelector/MultiSelector.
    choices:                        # Content of the specified option.
      - Correct
      - Discard
      - Questionable
```

©2013-2026 Tencent Cloud. All rights reserved.



In this scenario, you can configure a schema to display the non-editable Question field to prevent misoperations during the labeling process. You can also set the Answer field to an editable field, and set custom filtering enumeration values to Correct, Discard, and Questionable.

©2013-2026 Tencent Cloud. All rights reserved.

Creating a Data Labeling Task for Traditional CV

Last updated: 2026-02-11 18:25:11

Overview

The Data Center > Data Labeling module allows you to create labeling tasks for image classification, object detection, object tracking, and image segmentation under Traditional CV Modeling, providing you with a professional and efficient labeling console.

> **Note:**
> A user can maintain up to 50 labeling tasks across all regions in the public cloud simultaneously. If you have expansion needs for your business, you can submit a ticket to contact us.

Creating a Data Labeling Task

The following describes how to create and complete a data labeling task.

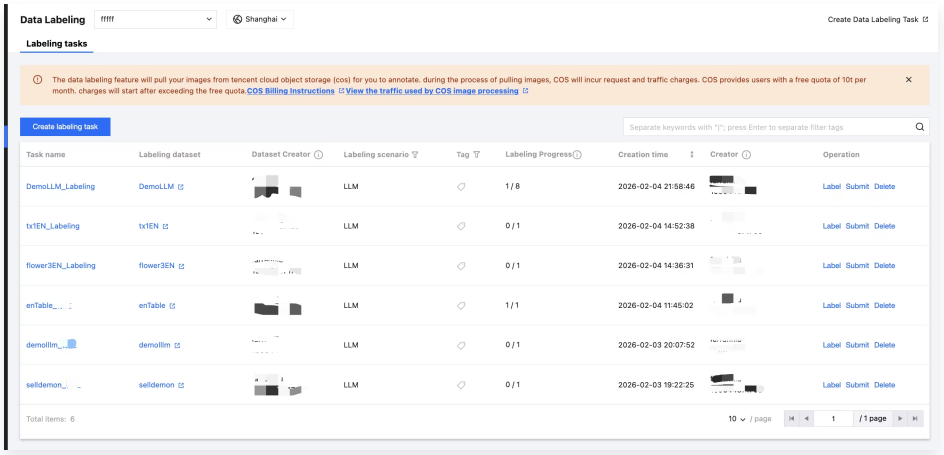
1. Prerequisites: You have successfully imported a dataset of the image type in the Data Center > Datastes module. The dataset has been imported successfully and is already in the Available status.
2. Click Create labeling Task in the **Data Center > Data Labeling** module and complete the parameter configuration.
   - Task Name: You can enter a custom labeling task name.
   - Task Description: You can enter an additional description for this task.
   - Tag: You can add the relevant tag information for the labeling task. For details about the tag feature, see Documentation of the Tag Product of Tencent Cloud.
   - Select Dataset: This drop-down list displays the datasets of the image type in the Available status in Data Center > Datasets. The user who creates this labeling task should have write permissions to the COS path corresponding to the dataset.
   - Labeling Scenario: You can select the corresponding labeling scenario. Currently, only labeling for the data of the image type is supported. Labeling for the data of the text type and video type is coming soon.
   - Labeling Format: You can select the corresponding labeling format based on the actual labeling tion scenario. If the labeling scenario supports only one format, it will be automatically selected by default on the backend. The Optical Character Recognition (OCR) scenario is currently available only to allowlisted users.
     - Image Classification

©2013-2026 Tencent Cloud. All rights reserved.

        - TI-ONE format
        - Structure of the file directory
    - Object Detection
        - TI-ONE format
        - Pascal Voc
        - COCO
    - Image Segmentation
        - TI-ONE format
        - Pascal Voc
        - COCO
    - Object Tracking
        - TI-ONE format
    - OCR
        - TI-ONE format
  - Tag Value: You can enter a custom tag value. You can also select the corresponding rendering color.
3. Click **Submit** on the configuration page. On the labeling task list page that appears, choose Operation > Label to go to the labeling console to label the image after the status of the labeling task becomes Created. You can access the same labeling task multiple times to perform the labeling operation.
4. During the labeling process, click Save to save your labeling results in real time. After you save the results and exit the current page, when you go to the console again, the results of your previous operation will be automatically recorded.
   Note: At this point, the labeling results have not yet been permanently written into the specified COS output path of the dataset.
5. On the labeling task list page, view the progress of the labeling task in real time through the Labeled status field. After all images in the dataset are labeled, you can choose **Operation > Submit** to submit the labeling results to your associated COS output path. Thereafter, other modules of Tencent Cloud TI-ONE Platform (TI-ONE) can use the labeling nformation of this

©2013-2026 Tencent Cloud. All rights reserved.

dataset.



©2013-2026 Tencent Cloud. All rights reserved.

Operations on the Data Labeling List

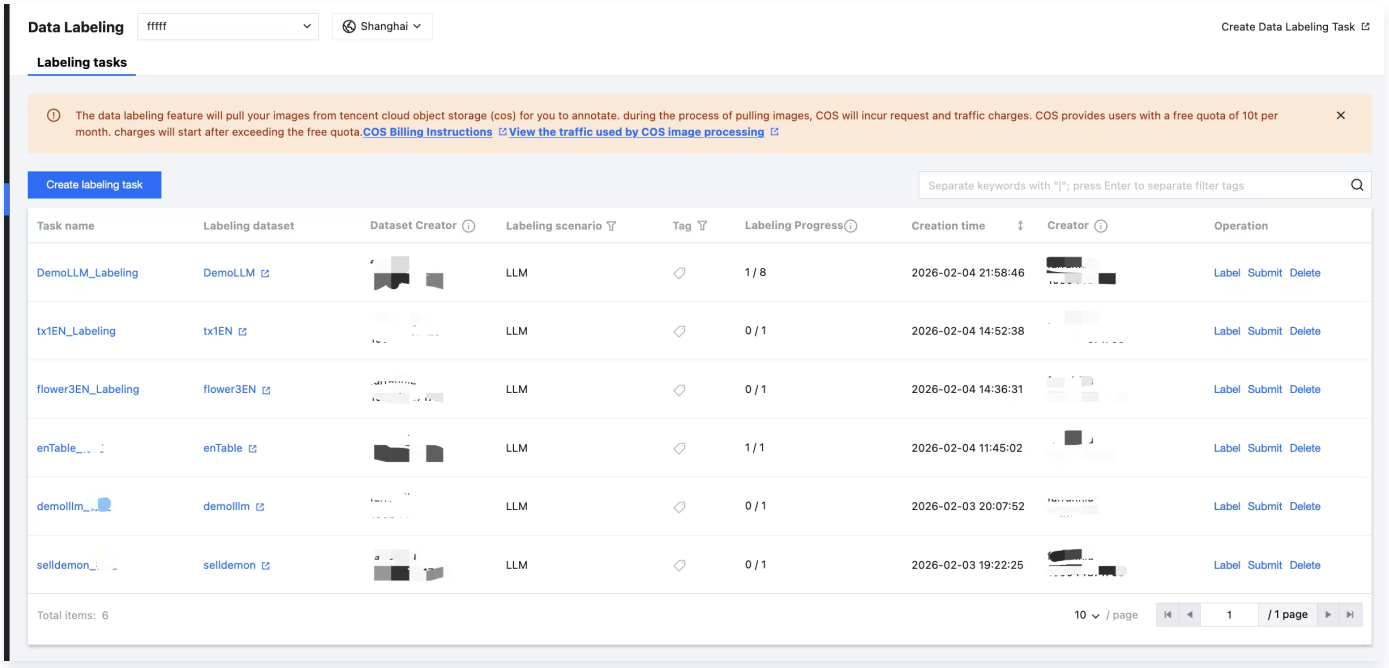
Last updated: 2026-02-11 18:24:33

Overview

The data labeling task list page centrally manages the collection of all types of labeling tasks under a user account, supporting the labeling, submitting, and deleting operations. It allows you to click an labeling task name to view the task details.

Introduction to Operations on the List Page

Operations on the data labeling task list page are described as follows:



- Task Name: displays the name of the labeling task. You can click the name to view the basic information of the task, such as the labeling scenario, labeling dataset, and labeling value collection.
- Labeling Dataset: displays the name and version number of the labeling dataset bound to the task.
- Labeling Scenario: displays the scenario category of the task. You can filter scenarios for display in the list.
- Tag: Tencent Cloud tag bound to the task.
- Status: The statuses include Creating, Created, and Creation Failed. You can filter statuses for display in the list.
- Labeled: displays the labeling progress of the current task, showing the total amount of labeled data or the total amount of data to be labeled.
- Creation Time: records the system time when the task was successfully created.

©2013-2026 Tencent Cloud. All rights reserved.

- Label: Click it to go to the corresponding labeling workbench to perform the labeling operation. You can access the same labeling task multiple times to perform the labelingon operation.
- Submit: Only after the information in the labeling workbench is manually submitted, the labeling result file can be submitted from the backend database of the system to the specified Cloud Object Storage (COS) output path and can be successfully read by other business modules of TI-ONE.
- Delete: You can delete the labeling task. After the task is deleted, the output labeling file generated by this task in the COS bucket will not be automatically deleted along with the task.

©2013-2026 Tencent Cloud. All rights reserved.

Labeling Workbench for Traditional CV Data

Last updated: 2026-02-11 18:26:34

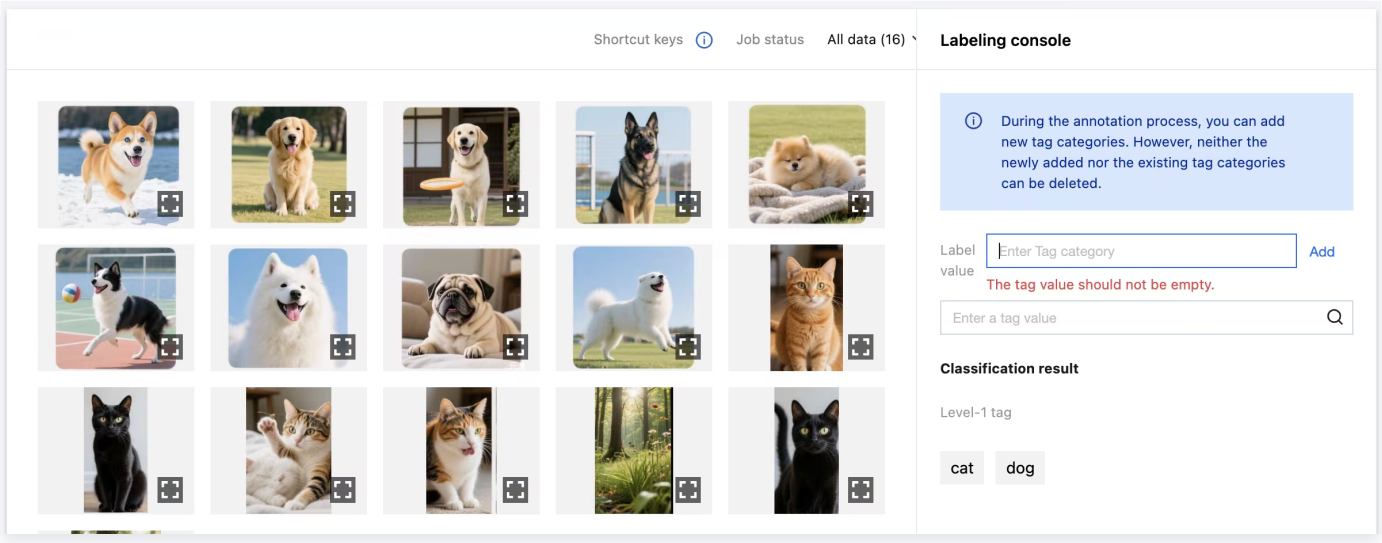
Overview

The Data Labeling functional module provides a professional and efficient labeling workbench. The different presentation forms of the labeling workbench are introduced as follows:

1. Labeling Workbench for image classification used for image classification labeling tasks.
2. Labeling Workbench for detection types such as object detection, object tracking, and image segmentation.
3. Labeling Workbench for Optical Character Recognition (OCR) types such as recognition/detection and intelligent structured labeling.

Labeling Workbench for Image Classification

The workbench uses a tiled small image list mode for the classification scenario, enabling you to quickly categorize multiple images.
In the upper-right corner of the page, you can filter images by job status: All Data, With Labeling Results, or Without Labeling Results.
The label value collection area is located on the right side of the workbench, allowing you to perform a search and assign label values to the currently selected images.

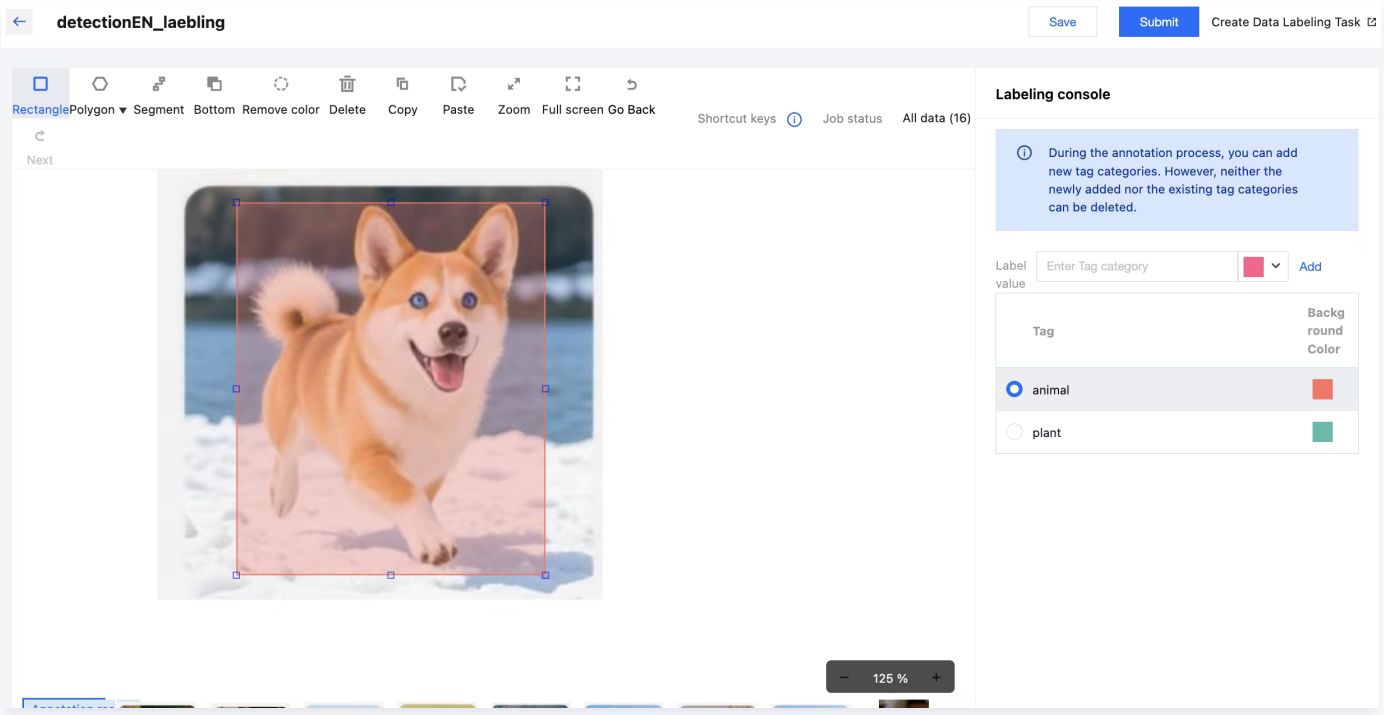


List of supported shortcuts:
Previous image: letter A on the keyboard;
Next image: letter D on the keyboard.

Labeling Workbench for Detection Types

©2013-2026 Tencent Cloud. All rights reserved.

The workbench uses a mode with a large image plus a row of thumbnail lists for the detection type scenario, enabling you to clearly identify and draw the positions of specified objects in images.
In the upper-right corner, you can filter images by job status: All Data, With Labeling Results, or Without Labeling Results.
The label value collection area is located on the right side of the workbench, allowing you to assign label values to the selected objects. At the top of the workbench, there is a collection of all labeling tools supported by the current labeling task: rectangle, polygon, polyline, and so on.



List of labeling tools:

- Rectangle: Draw a rectangular box (Ctrl+1).
- Polygon: Draw a polygonal box by plotting points with the left mouse button and ending with the right mouse button. You can specify a fixed number of polygonal points you can draw (Ctrl+2).
- Polyline: Draw line segments by plotting points with the left mouse button and ending with the right mouse button (Ctrl+3).
- Send to Back: Place the drawn boxes or line segments behind all other objects on the page (Ctrl+4).
- Remove Color: Cancel the default fill color of the box area, making the labeling area transparent and allowing the image content within the labeling area to be more visible.
- Delete: Delete the selected drawn boxes or line segments (Ctrl+D).
- Copy/Paste: Copy and paste the drawn boxes or line segments. This operation will also copy the label value attributes of the objects (Ctrl+C/Ctrl+V).
- Zoom: Currently, only polygons and polylines are supported for zooming.

©2013-2026 Tencent Cloud. All rights reserved.

- Full Screen: Click it to view an image in full-screen mode.
- Undo/Redo: Undo the previous or next action.
- List of shortcut keys: (1) Previous image: letter A on the keyboard; (2) Next image: letter D on the keyboard.

Labeling Workbench for OCR Types

Based on the two labelingotation types (recognition/detection and intelligent structured labeling) in the OCR labeling scenario, Tencent Cloud TI-ONE Platform (TI-ONE) provides a dedicated labeling workbench and a self-developed structured labeling language called descriptor.

Introduction to Labeling Tools and Shortcut Keys

During the labeling process, you can click **Use Instructions** in the upper-right corner of the labeling workbench to view details.

©2013-2026 Tencent Cloud. All rights reserved.

Note

Supported checkbox (Copy and Paste Into the Field Text for Use)

☐☑☒

Annotation Tool

Annotation Tool	Auto	kv [redacted]	T [redacted]	Point	ID [redacted]	... [redacted]	[redacted]	Box	AUTO [redacted]
Description	Full-Image Pre-Labeling	Show field attributes under the annotation box.	Display field text under the annotation box.	Show the four-point sequence in the annotation box.	Show id under the annotation box.	Adjust the font and size for display.	Automatically identify field text after framing.	Identify the field text of the selected description box.	Automatically save when paging.

3. Shortcut Key

Macos Shortcut Keys	Win shortcut key.	Shortcut Key Description
Command+mouse	Ctrl+mouse	Move background image
Option + mouse scroll wheel.	Alt + scroll wheel.	Zoom in and out of the image.
F1	F1	Copy box
F2	F2	Modify Field Content
F3	F3	Modify field properties.
F4	F4	Delete Selected box
F5/Ctrl+S	F5/Ctrl+S	Save

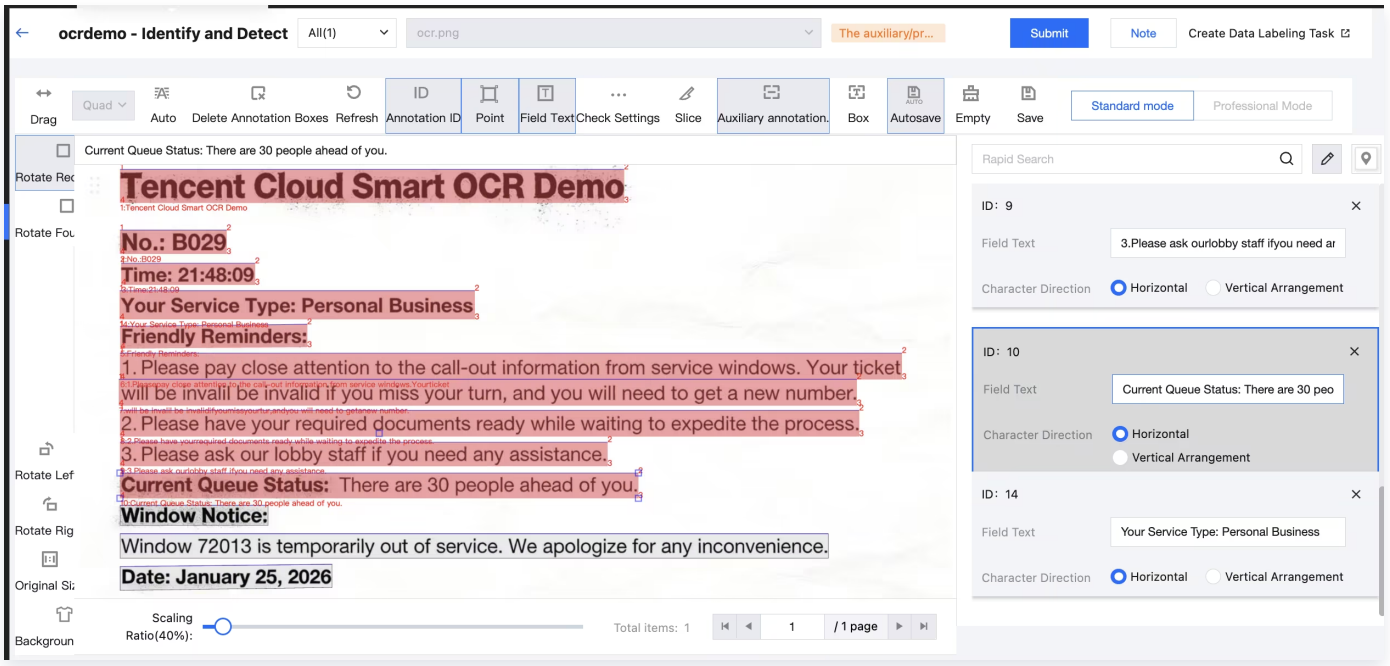
Introduction to Labeling Modes

The tagging area on the right side of the labeling workbench provides two modes: standard mode and professional mode. You can switch between the two modes.

- Standard mode
  A simplified labeling mode is provided for both recognition/detection and intelligent structured labeling types. In this mode, you can directly decide the direction (horizontal/vertical) of text detection and recognition by clicking the buttons on the frontend page, or pair the key and value in structured data by clicking relevant buttons.

©2013-2026 Tencent Cloud. All rights reserved.

- Professional mode
  A professional labeling mode is provided for the intelligent structured labeling type. In this mode, you can enter the self-developed descriptor language to adapt to complex text scenarios under the structured OCR. The descriptor language is described as follows:



Introduction to the Descriptor Language

Intelligent structured labeling outputs are primarily key-value pairs (KV), for example, invoice code (key) -- 141001950242 (value) or invoice number (key) -- 30148556 (value). Therefore, the most important attribute in field attributes is the KV attribute. In addition, in actual OCR business scenarios, there are often complex and diverse requirements, such as nested KV pairs, incomplete fields, fields across lines, additional key fields, or normalization of key fields. Therefore, TI-ONE has developed the Descriptor labeling language to flexibly adapt to various OCR scenarios.

KV Attribute: K-Descriptor and V-Descriptor

This attribute can describe whether a field is a key or a value (K field and V field, respectively), and their KV association relationships. A K field and a V field can exist independently, or they can exist together and are associated with each other. There are several types of KV association relationships:

- 1k0v: indicates that there is only a K field, without an associated V field.
- 0k1v: indicates that there is only a V field, without an associated K field.
- 1k1v: indicates that a K field is associated with a V field. Both the K field and V field appear on an image.
- 1k1v: indicates that only a V field appears in an image. However, this V field needs to be associated with an additional key name. The additional key name is identified by using @@ in

©2013-2026 Tencent Cloud. All rights reserved.

the field text.

- 1knv: indicates that a K field is associated with multiple V fields, which often occurs when these V fields are split across multiple lines.
- nk1v: indicates that multiple K fields are associated with a V field, which often occurs in nested tables or row-column tables.

Example: enter "k3,5": indicates that field 3 and field 5 form a KV relationship, where 3 is a key box and 5 is a value box.

Completeness (complete) Attribute: C-Descriptor

This attribute describes whether a field fully expresses a meaning. If it can fully express a meaning, it is called a complete field. When multiple fields need to be combined, the order of the combination must be described. In most cases, a single field can express a complete meaning. However, sometimes due to reasons such as line breaks or a vertical text layout, multiple fields must be connected in order to convey a complete meaning.
Definitions of letter c followed by the following formats of numbers:

- 0: This field is complete. In this case, the c-descriptor is optional.
- fid1, fid2...: fid represents the ID number of each box, which indicates that this field is not complete. Multiple fields separated by commas in a sequence must be combined to convey a complete meaning.

Example: The descriptor for field 5 is entered as: 5kc5,7; the descriptor for field 7 is entered as: 7kc5,7. This means that fields 5 and 7 must be combined to convey a complete meaning.

©2013-2026 Tencent Cloud. All rights reserved.

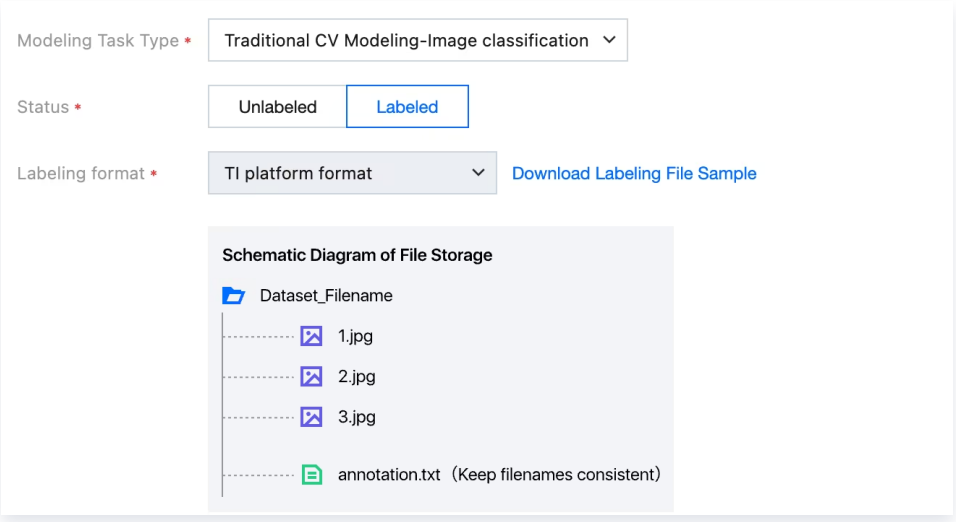
Tag Format Examples

Last updated: 2026-02-11 18:25:24

Overview

When you need to upload an labeled dataset of the image or text type in the **Data Center** > **Datasets** module of Tencent Cloud TI-ONE Platform (TI-ONE), the format of the labeled file content must comply with platform specifications. Only labeled files that meet the specifications can be correctly parsed by the backend. After successful parsing, you can view the visual distribution of labeled information for this dataset on the dataset details page.
In addition, we recommend that the size of the labeled content per image not exceed 100 KB. To perform a scale-out, submit a ticket to contact us.
When you import an labeled dataset, ensure that it meets the following two requirements:

- You need to prepare your file directory according to the "Schematic Diagram of File Storage".
- You need to prepare your labeled file according to the content format described in the "Download Labeling File Sample" link.



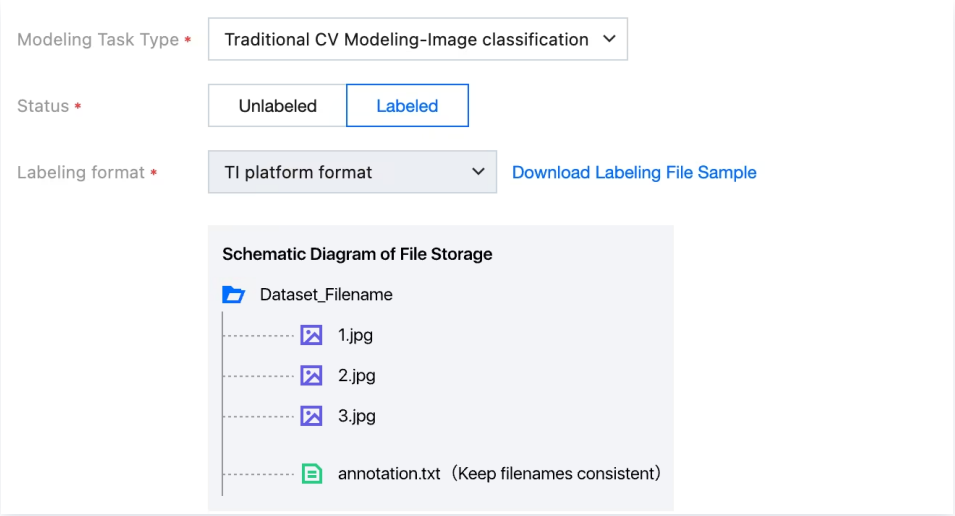
The following section introduces the labeled format specifications for different label scenarios.
Note: The file size upper limit for the labeled content per image is 1 MB.

Image Classification

(1) TI-ONE Format

- Requirements for the file directory:

©2013-2026 Tencent Cloud. All rights reserved.



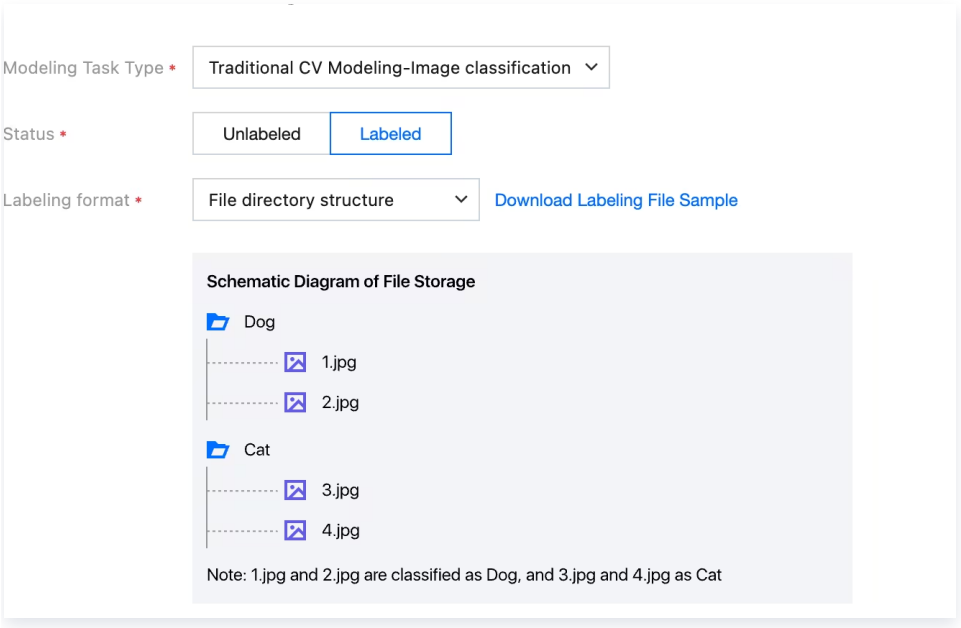
- Requirements for the labeled file format:

```
{
    "info": { // An optional field, indicating the image-related
information.
        "md5": "MD5 value of an image", // An optional field, indicating
the MD5 value of an image.
        "path": "Relative path of an image.",
    },
    "tags": {
        "classification_tags": [{
            "first_class": "Value of the first-level tag.",
            "confidence_level": Confidence // An optional field,
indicating the tag confidence.
        }]
    }
}
```

(2) Structure of the File Directory

- Requirements for the file directory:

©2013-2026 Tencent Cloud. All rights reserved.

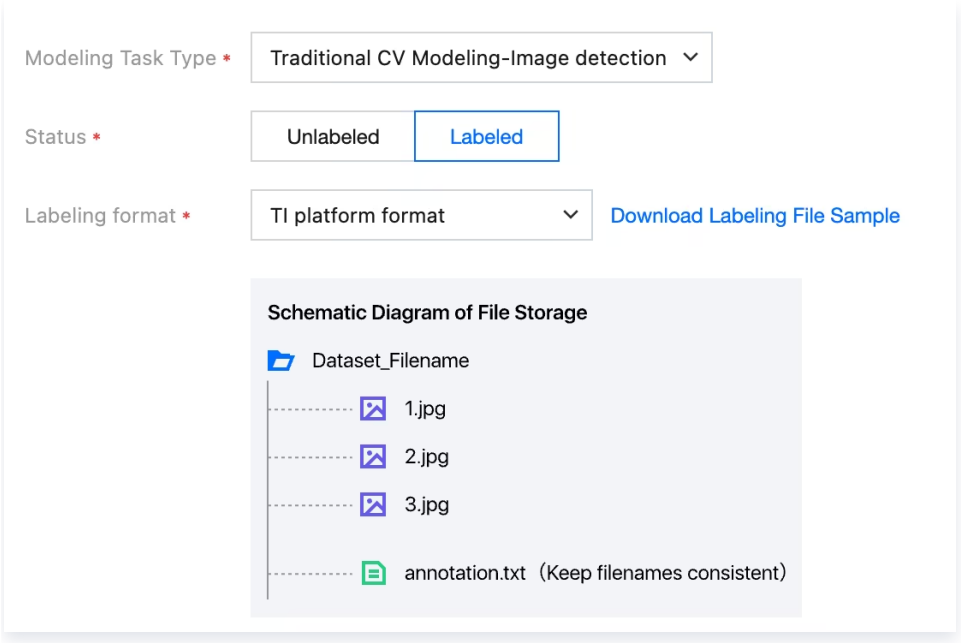


- Requirements for the labeled file format: You do not need to upload an additional labeled file.

Object Detection

(1) TI-ONE Format

- Requirements for the file directory:



- Requirements for the labeled file format:

```
{
    "tags": {
        "detection_tags": {
            "shapes": [{
```

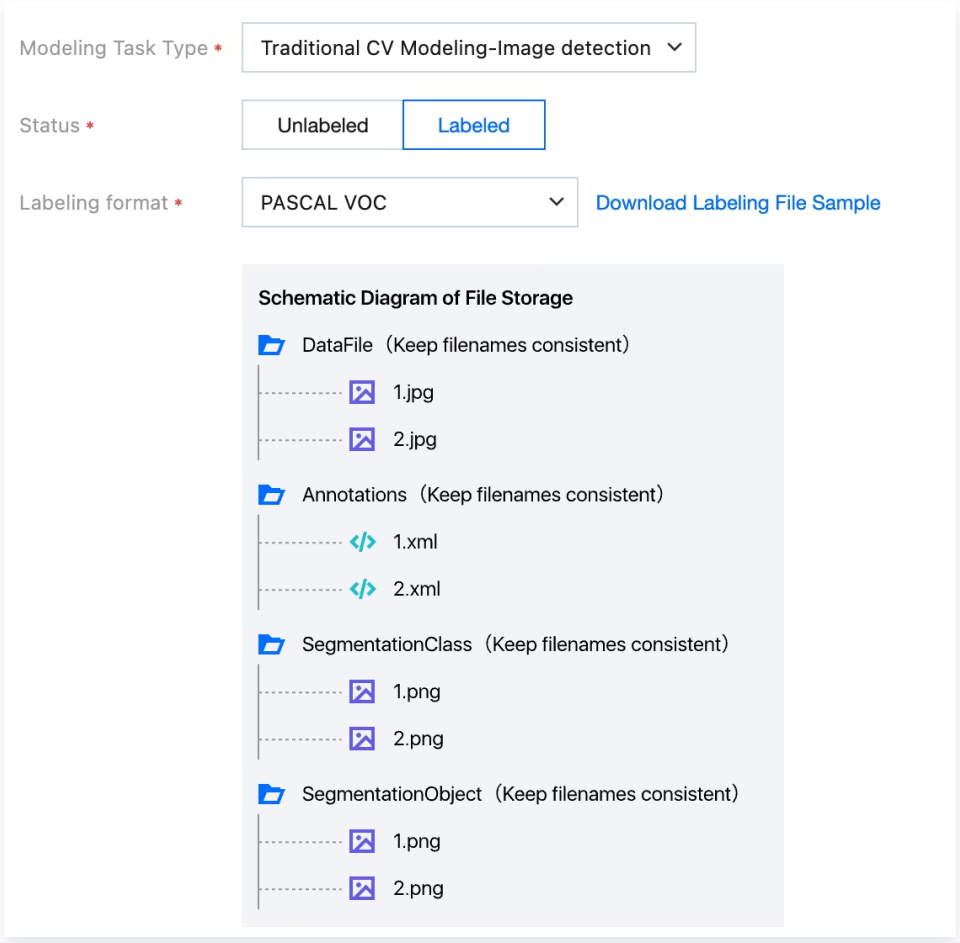
©2013-2026 Tencent Cloud. All rights reserved.

```
                "frame_type": "frame shape", // Indicating the shape of
the frame. Polygon, rectangle, and polyline are supported.
                "pts": [x1, y1, x2, y2...],
                "bbox": [Horizontal coordinate in the upper-left corner,
vertical coordinate in the upper-left corner, width, height],
                "tag": [{
                    "label_key_name": "Tag field",
                    "label_value_name": ["Tag value"],
                    "confidence_level": [Confidence] // An optional field,
indicating the tag confidence.
                }]
            }]
        }
    },
    "info": { // An optional field, indicating the image-related
information.
        "md5": "MD5 value of an image", // An optional field, indicating
the MD5 value of an image.
        "path": "Relative path of an image."
    }
}
```

(2) Pascal Voc

©2013-2026 Tencent Cloud. All rights reserved.

- Requirements for the file directory:



- Requirements for the labeled file format: (XML file format)

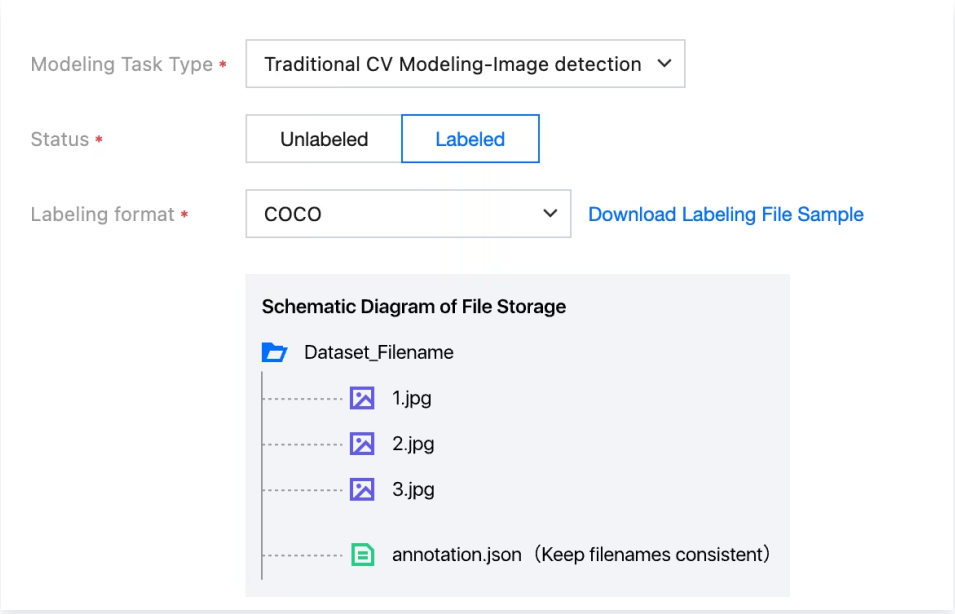
```
<annotation>
    <folder>VOC2012</folder>
    <filename>2007_000033.jpg</filename>
    <source>
        <database>The VOC2007 Database</database>
        <annotation>PASCAL VOC2007</annotation>
        <image>flickr</image>
    </source>
    <size>
        <width>500</width>
        <height>366</height>
        <depth>3</depth>
    </size>
    <segmented>0</segmented>
    <object>
        <name>aeroplane</name>
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        <pose>Unspecified</pose>
        <truncated>0</truncated>
        <difficult>0</difficult>
        <bndbox>
            <xmin>9</xmin>
            <ymin>107</ymin>
            <xmax>499</xmax>
            <ymax>263</ymax>
        </bndbox>
    </object>
</annotation>
```

(3) COCO

- Requirements for the file directory:



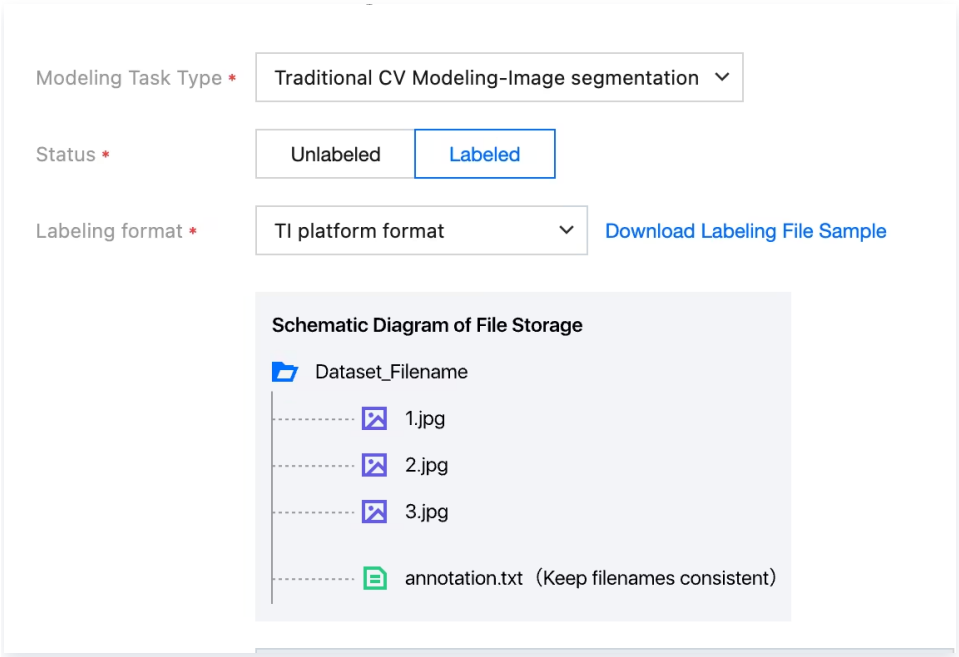
- Requirements for the labeled file format: You need to strictly refer to Object Detection formats on the official website of COCO .

Image Segmentation

(1) TI-ONE Format

- Requirements for the file directory:

©2013-2026 Tencent Cloud. All rights reserved.



- **Requirements for the labeled file format:**

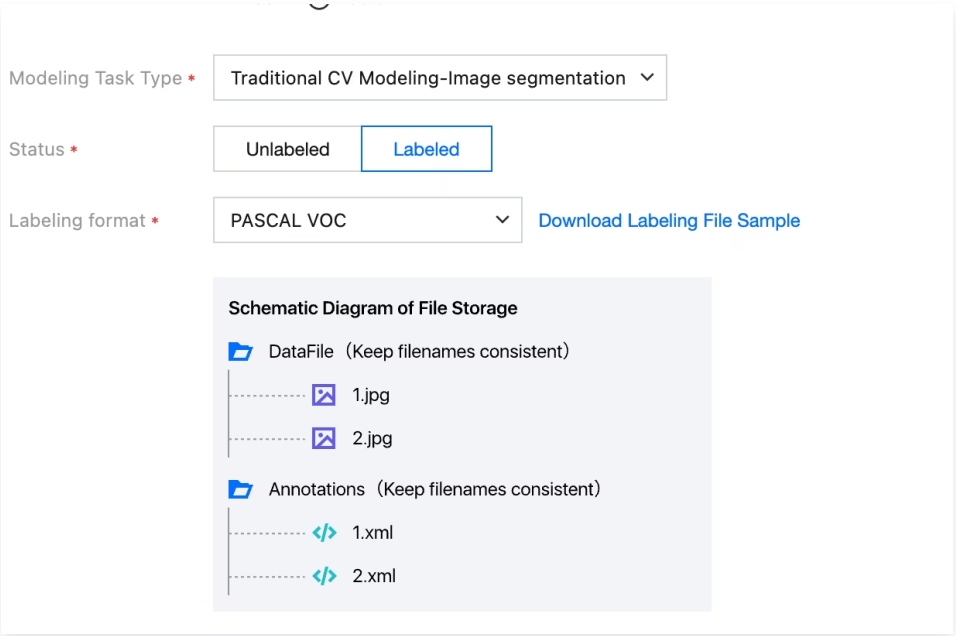
```
{
    "tags": {
        "segmentation_tags": { //[Note] When you import information to the
annotation platform, either shapes or mask should be provided. If both are
provided, shapes will be used by default. When you export information from
the annotation platform, both shapes and mask will be provided;
            "shapes": [{
                "frame_type": "frame shape", // Indicating the shape of
the frame. In public cloud scenarios, polygon is supported, and this field
is required.
                "pts": [
                    [x1, y1, x2, y2...]
                ],
                "bbox": [Horizontal coordinate in the upper-left corner,
vertical coordinate in the upper-left corner, width, height], //Optional.
pts represents the bounding rectangle box of a segmentation area.
                "tag": {
                    "label_value_name": "Tag value",
                    "confidence_level": Confidence // An optional field,
indicating the tag confidence.
                }
            }],
            "mask": {
                "path": "Relative path of the mask image.",
```

©2013-2026 Tencent Cloud. All rights reserved.

```
                "notes": {
                    "Grayscale value": "Tag value"
                }
            }
        }
    },
    "info": { // An optional field, indicating the image-related
information.
        "md5": "1e81a232952f32af71b70848556e0ecd", // An optional field,
indicating the MD5 value of an image.
        "path": "Relative path of an image."
    }
}
```

(2) Pascal Voc

- Requirements for the file directory:



- Requirements for the labeled file format: (requirements for the XML file format)

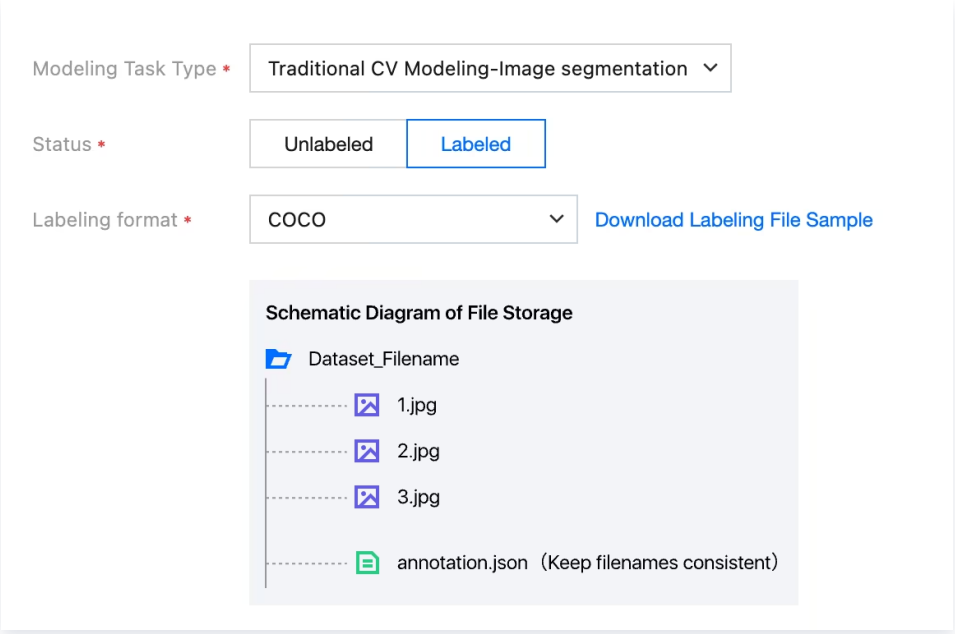
```
<annotation>
      <folder>VOC2012</folder>
      <filename>2007_000039.jpg</filename>
      <source>
          <database>The VOC2007 Database</database>
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        <annotation>PASCAL VOC2007</annotation>
        <image>flickr</image>
    </source>
    <size>
        <width>500</width>
        <height>375</height>
        <depth>3</depth>
    </size>
    <segmented>1</segmented>
</annotation>
```

(3) COCO

- Requirements for the file directory:



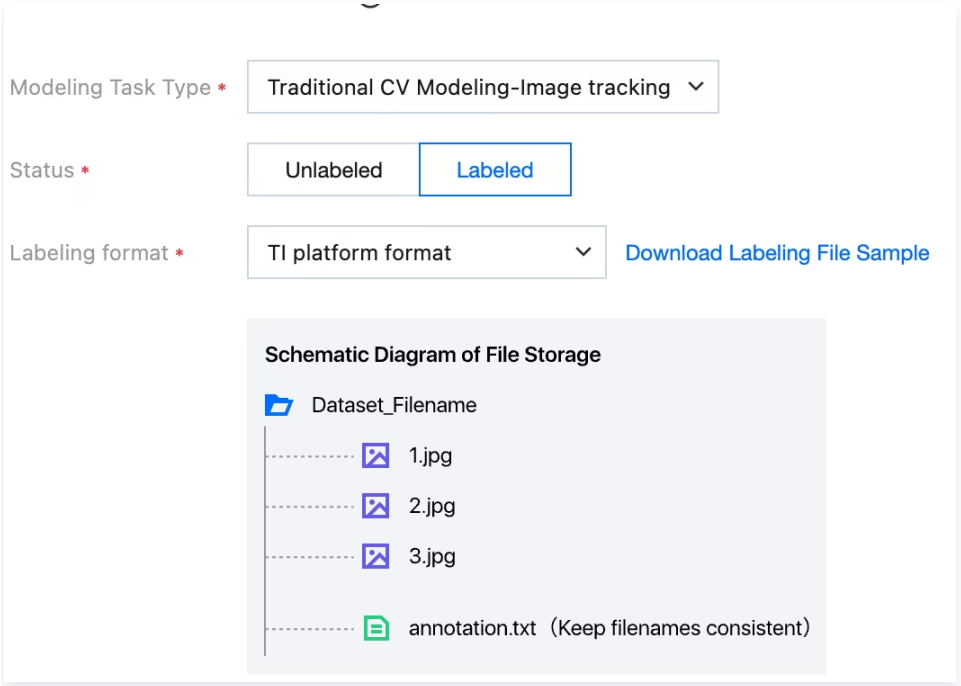
- Requirements for the labeled file format: You need to strictly refer to Object Detection formats on the official website of COCO .

Object Tracking

TI-ONE Format

- Requirements for the file directory:

©2013-2026 Tencent Cloud. All rights reserved.



- **Requirements for the labeled file format:**

```
{
  "tags": {
      "detection_tags": {
          "shapes": [{
              "frame_type": "frame shape", // Indicating the shape of
the frame. Polygon, rectangle, and polyline are supported.
              "pts": [x1, y1, x2, y2...],
              "bbox": [Horizontal coordinate in the upper-left corner,
vertical coordinate in the upper-left corner, width, height],
              "tag": [{
                  "label_key_name": "Tag field",
                  "label_value_name": ["Tag value"],
                  "confidence_level": [Confidence] // An optional field,
indicating the tag confidence.
              }]
          }]
      }
  },
  "info": { // An optional field, indicating the image-related
information.
      "md5": "MD5 value of an image", // An optional field, indicating
the MD5 value of an image.
      "path": "Relative path of an image."
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    }
}
```

Text Classification

For details about the organization form of text data and labeled information in the text classification scenario, see the prompt cards on the dataset import frontend page.



In addition, the following further explains the tag formats that can be flexibly extended in TI-ONE Format and JSON Format (similarly for the line-based format and file-based format):

```
- First-level single-choice format
"Sentiment": "Negative"
"Sentiment": "Positive"

- First-level multi-choice format
"Fields": ["Politics","Economy"]
"Fields": ["Entertainment","Society"]
```

©2013-2026 Tencent Cloud. All rights reserved.

Data Construction

Last updated: 2026-02-11 18:25:46

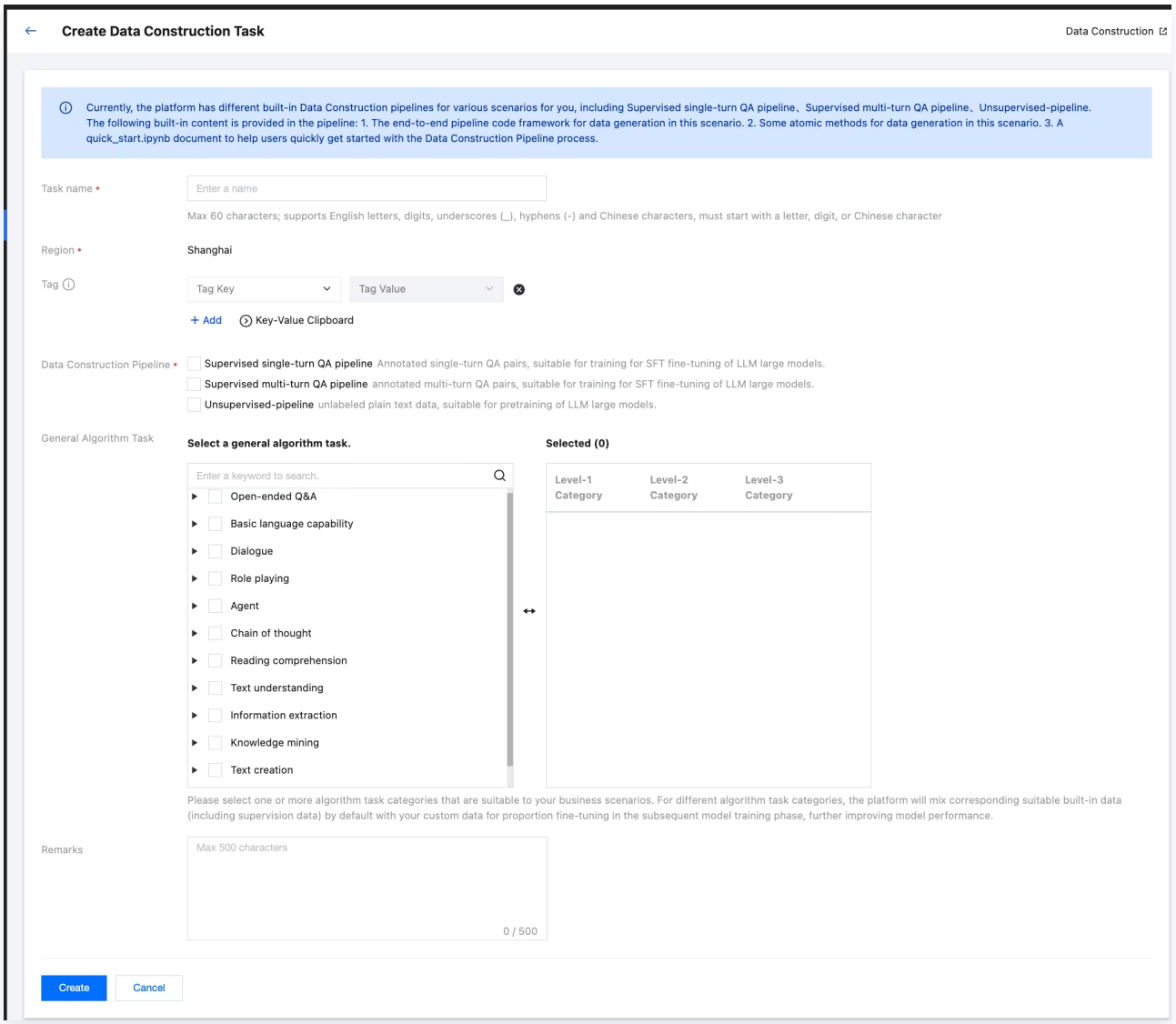
Overview

High-quality training data is crucial for improving the fine-tuning effectiveness of models. To help you prepare high-quality training data, Tencent Cloud TI-ONE Platform (TI-ONE) has accumulated the standard data preprocessing pipelines used in internal R&D into the **Data Center > Data Construction** module. The platform provides three types of built-in data processing pipelines for different scenarios: supervised single-turn Q&A pipeline, supervised multi-turn Q&A pipeline, and unsupervised pipeline. In addition, each pipeline provides specific features based on different business requirements, including data cleaning, prompt optimization, data filtering, and data augmentation.

Creating a Task

1. Choose **Data Center > Data Construction**, and click **Create Task** on the list page to configure the parameters:
   - Task Name: a user-defined input. The name should be globally unique in the **Data Construction** module.
   - Tag: a unified cloud resource tag of Tencent Cloud. It is used to categorize and manage resources from different dimensions. For details, see the official document of Tag.
   - Data Contruction Pipeline: Multiple pipelines can be selected. Their types include:
     - Supervised Single-Turn Q&A Pipeline: a single-turn Q&A pair with annotations. It is suitable for supervised fine-tuning (SFT) training of LLMs.
     - Supervised Multi-Turn Q&A Pipeline: a multi-turn Q&A pair with annotations. It is suitable for SFT training of LLMs.
     - Unsupervised Pipeline: plain text data without annotations. It is suitable for LLM pretraining.
   - General Algorithm Tasks: You can select one or more algorithm task types that closely match your actual business scenarios. For different algorithm task types, during the subsequent model training phase, the platform will automatically match the appropriate built-in data (supervised data) to mix with your custom data to achieve proportioned fine-tuning, further improving the model effectiveness.

©2013-2026 Tencent Cloud. All rights reserved.



2. Click **Submit** on the configuration page. The system will return to the data construction task list page:
   - Creation Time: time when the data preprocessing task was created.
   - Operations:
     - Navigate to Corresponding Dev Machine: Go to the Dev Machines module. After the dev machine is successfully created, the relevant information will be displayed synchronously in the Corresponding Dev Instance ID column.
     - Delete: If the corresponding dev machine has been deleted, you can directly delete the task. If the corresponding dev machine has not been deleted, you need to delete the dev machine first and then the data construction task.
   - Search box: You can perform a fuzzy search by **Task Name**.

Redirecting to the Corresponding Dev Machine

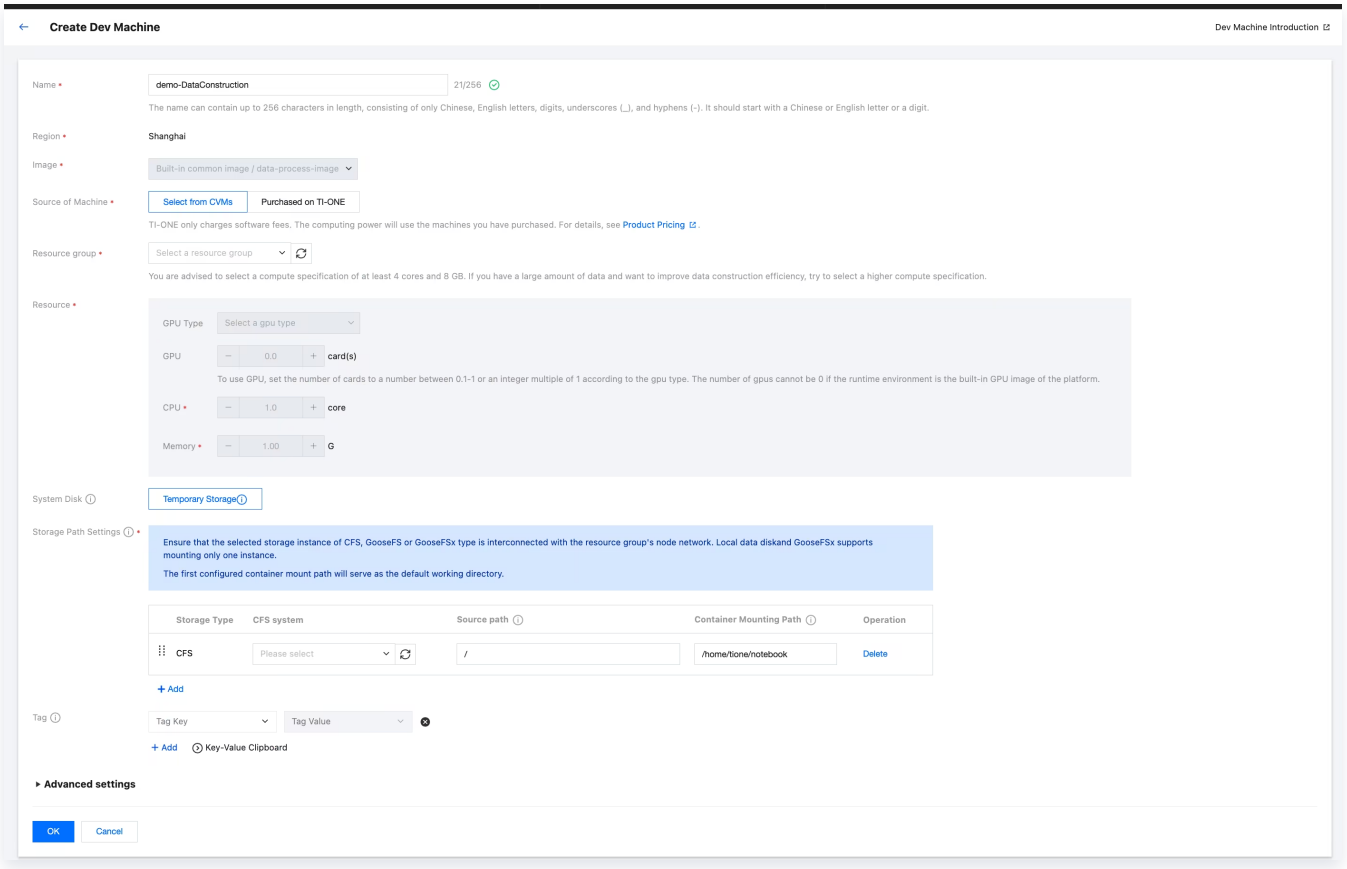
©2013-2026 Tencent Cloud. All rights reserved.

Choose **Data** Construction> **Operations > Navigate to Corresponding Dev Machine** to go to the Dev Machines module.

1. If you click Navigate to Corresponding Dev Machine for the first time for a task, you need to create a dev machine first.

   When you click **Navigate to Corresponding Dev Machine** for the first time for a task, you need to create a dev machine. After you click the dev machine, the platform will automatically redirect to the **Training Workshop** > **Dev Machines** > **Create Dev Machine** parameter configuration page:

   - Image: If you are redirected to the current page from the **Data Construc** module, the dedicated built-in image of Data Construction is selected by default and cannot be modified. This image is displayed only when you are redirected to the current page from the Data Construction module. If you are redirected to the current page from the general entry of the Dev Machines module, you cannot find the new built-in image.
   - CVM instance specifications: Regular data construction tasks require CPU only and do not need GPU.
   - Storage configuration: Currently, the recommended storage is Cloud File Storage (CFS).
   - Other parameter configurations should be consistent with the existing features of the dev machine.

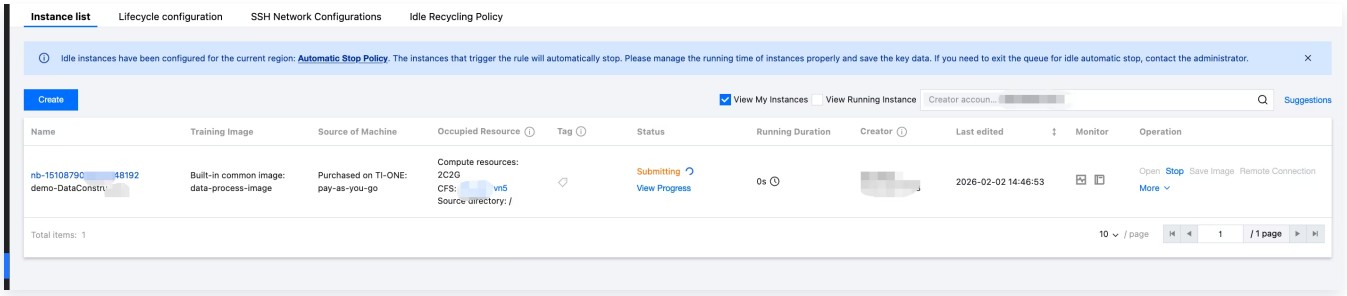


Click **OK** on the configuration page. After the task is created, the system returns to the dev machine list page to wait for the dev machine to be successfully created.

©2013-2026 Tencent Cloud. All rights reserved.

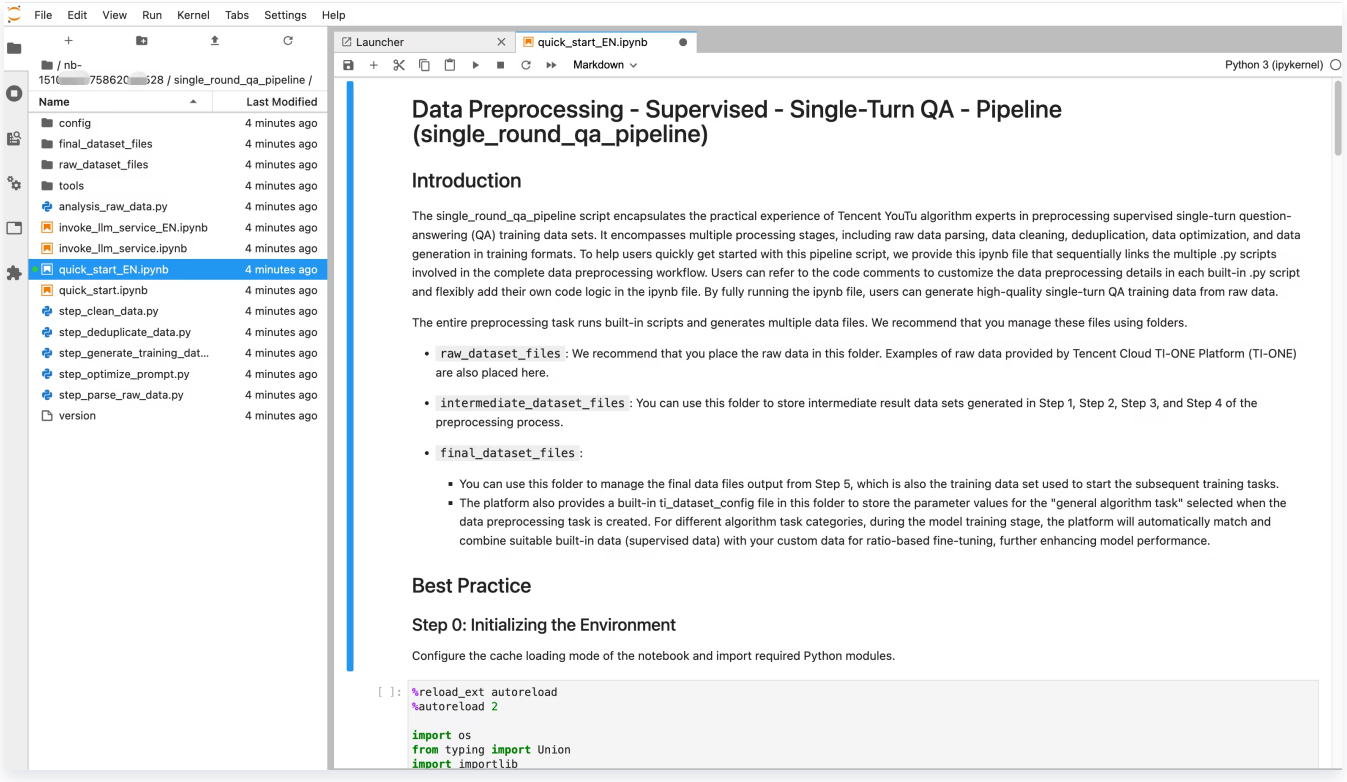
2. If you are not clicking Navigate to Corresponding Dev Machine for the first time for a task, you will be redirected to the dev machine list.

   After a dev machine is successfully created, if you choose **Data Construction** > **Navigate to Corresponding Dev Machine** again, you will be redirected to the dev machine list page. By default, the corresponding dev machine ID will automatically appear in the search box of the list to display the search results.



3. Access the dev machine to edit the preset pipeline script.

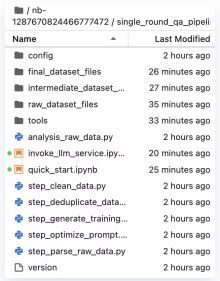
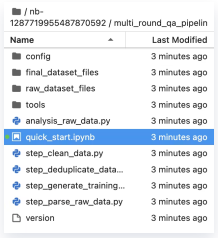
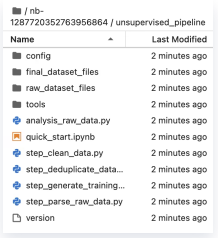
   The preset data preprocessing pipelines of the platform are displayed in the dev machine by default. Since the Data Preprocessing Pipeline parameter on the parameter configuration page for creating a task allows you to select multiple pipelines, one or more built-in pipelines may exist in the dev machine:



   The following is built in each pipeline:
   a. An end-to-end data preprocessing pipeline code framework for the current scenario.
   b. Some atomic methods for data construction for the current scenario.
   c. A quick_start.ipynb file to help you quickly get started with the pipeline preprocessing process.

©2013-2026 Tencent Cloud. All rights reserved.

For details about the hierarchical structure of a folder, see the following:

single_round_qa_pipeline	multi_round_qa_pipeline	unsupervised_pipeline
		

Instructions on Raw Dataset Formats

Supervised Data Formats

Our supervised data preprocessing script code supports a wide range of common data formats for LLMs. Currently, the **supervised single-turn Q&A pipeline and supervised multi-turn Q&A pipeline** support the following formats:

1. **json file**: The JSON content is an array where each item is a valid JSON object that can be converted into single-turn Q&A data. If the JSON content is not an array but a JSON object, it will be parsed into an array with a length of 1 element for processing.
2. **jsonline file**: Each line is a valid JSON object that can be converted into supervised Q&A data.
3. **parquet file**: Parquet is a common, open-source, columnar data storage format (see Parquet Documentation). We will convert a Parquet object into a JSON object first. You can preview the JSON format in the following sampling process, and then perform data processing.
4. **Data set folder**: The folder contains multiple valid data files (their formats can be json, jsonline, or parquet). In this case, the folder will be processed as a dataset, and the dataset name is the folder name. After the subsequent data parsing process is completed, the contents of all files will be merged, and then the subsequent processes will be executed.
   - Supervised single-turn data format: The file format of the raw dataset provided in the sample code is **jsonl**. The sample content is as follows:

```
{"instruction": "Please briefly answer the following question:",
"input": "What investment and financial management methods do you
usually use?", "output": "I usually use bank savings, which have
relatively low risk."}
{"instruction": "Please briefly answer the following question:",
"input": "What is a fund split?", "output": "A fund split is a method
of recalculating fund assets by changing the relationship between the
```

©2013-2026 Tencent Cloud. All rights reserved.

```
net asset value per share and the total number of fund shares, while
remaining the total asset value of a fund investor unchanged."}
```

- Supervised multi-turn data format: The file format of the raw dataset provided in the sample code is **jsonl**. The sample content is as follows:

```
{"conversations": [{"from": "human", "value": "What is insurance? Is
it reliable?"}, {"from": "model", "value": "Insurance refers to a
commercial activity in which the policyholder pays premiums to the
insurer according to the contract, and the insurer is responsible for
compensating the insured for property losses caused by possible
accidents specified in the contract, or for paying insurance benefits
when the insured dies, becomes disabled, falls ill, or reaches the
age, deadline, or other conditions stipulated in the contract.
Insurance products purchased through formal channels are safe and
reliable."}, {"from": "human", "value": "I want to buy some
investment and financial management insurance products. What
insurance types are there?"}, {"from": "model", "value": "Annuity
insurance, dividend insurance, investment-linked insurance, and so
on"}, {"from": "human", "value": "Are these investment and financial
management insurances guaranteed to make profits without any
losses?"}, {"from": "model", "value": "No. The profits are affected
by market factors."}]}
{"conversations": [{"from": "human", "value": "My mother wants to buy
the critical illness insurance recently. Is this a scam?"}, {"from":
"model", "value": "No. Critical illness insurance is an insurance
product designed to provide coverage for serious illnesses."},
{"from": "human", "value": "What is the reimbursement ratio for this
critical illness insurance?"}, {"from": "model", "value": "The
reimbursement ratio for critical illness insurance is determined
based on the specific terms agreed upon in the insurance contract.
Different insurance companies and insurance products may have
different reimbursement ratios."}, {"from": "human", "value": "My
mother also wants to buy one for me, but I am currently 17 years old.
Can I buy it?"}, {"from": "model", "value": "Yes. However, minors
need their parents as policyholders to buy an insurance product."}]}
```

Unsupervised Data Formats

Our unsupervised data preprocessing script code supports a wide range of common data formats for LLMs. Currently, the **Unsupervised Pipeline** supports the following formats:

©2013-2026 Tencent Cloud. All rights reserved.

1. **json file**: supports parsing both JSON object files and JSON array files.
2. **jsonline file**: supports the jsonline multi-line file format.
3. **parquet file**: Parquet is a common, open-source, columnar data storage format (see Parquet Documentation). We first convert a Parquet object into a JSON object. You can preview the JSON format in the following sampling process, and then perform data processing.
4. **text file**: supports files in TXT format.
5. **wet format**: The WET file format refers to the Web ARChive (WARC) text format used on raw web pages. We will parse a WET file into a JSON object. You can preview the JSON object in the sampling process and then perform data processing.
6. **Data set folder**: supports placing multiple files of the above types in the same folder for processing. Currently, only multiple files of the same type in the same folder can be processed simultaneously.

Examples of data formats are as follows:

- json: Each line of a JSON array is regarded as a piece of data.

```
[
  "this is statement 1",
  "this is statement 2"
]
```

- jsonl: The value field corresponding to the text key in each line is regarded as a piece of data. For example:

```
{"text": "this is statement 1"}
{"text": "this is statement 2"}
```

- text:

```
this is statement 1
this is statement 2
```

- wet: supports the standard Web Archive format. For details, see Sample Data.

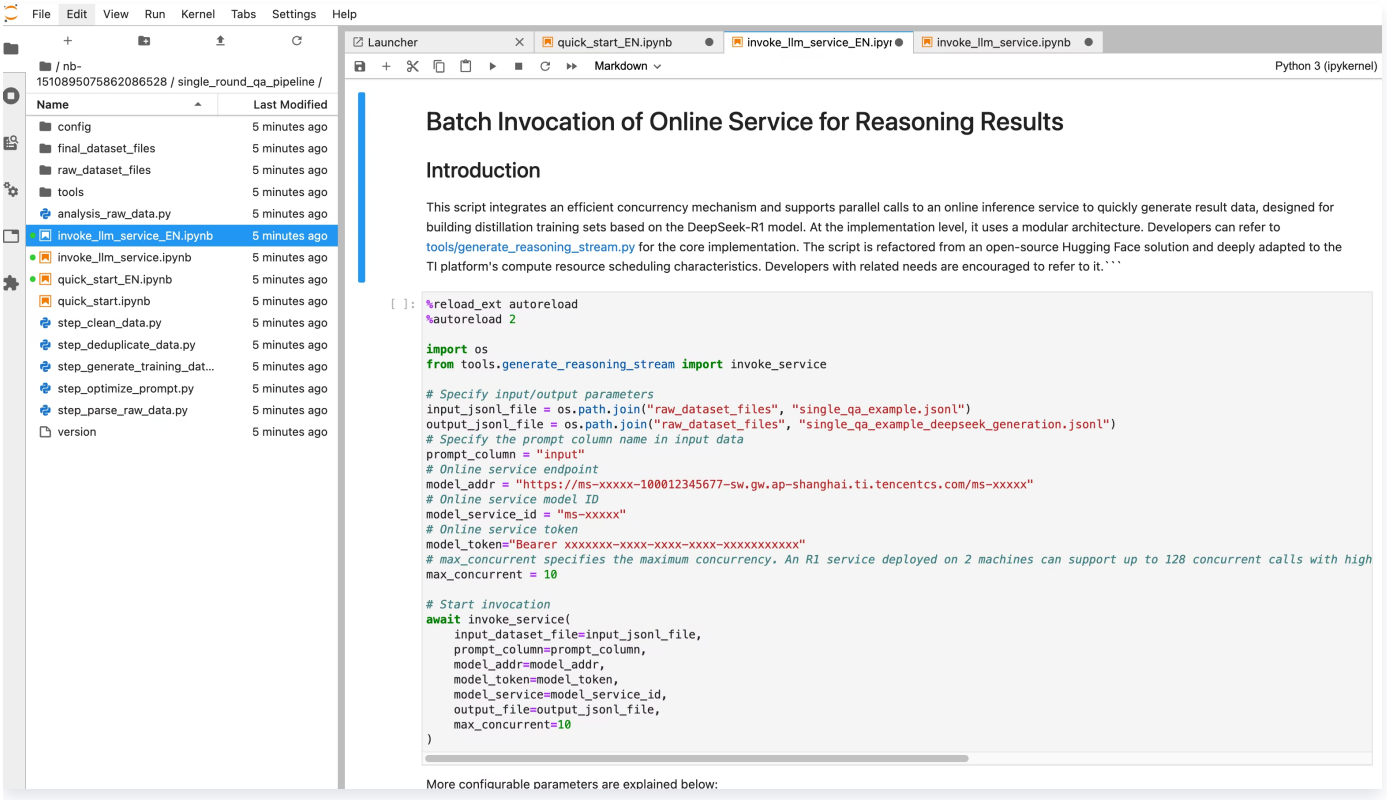
Data Distillation

In the dev machine script of the Supervised Single-Turn Q&A Pipeline, we have added a built-in data distillation script for obtaining batch prediction results by calling the Online Services module through the data preprocessing notebook. This script helps you achieve goals such as using the

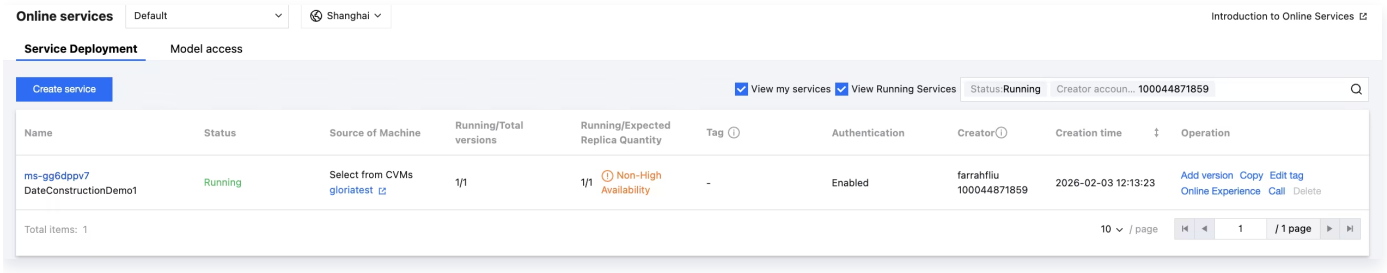
©2013-2026 Tencent Cloud. All rights reserved.

DeepSeek-R1 model to build a distilled training set. In addition, this data distillation step has been built into the existing data preprocessing pipeline, making it easier for you to flexibly process data. For example, you can filter out low-quality or redundant sample data through sub-modules in the pipeline, such as data cleaning and data deduplication, before calling the distillation model, thereby reducing the costs of service calls.

Developers can view the core implementation code through tools/generate_reasoning_stream.py. This script is rebuilt based on the Hugging Face open-source scheme, is deeply adapted to the computing resource scheduling feature of TI-ONE, and is integrated with an efficient concurrency mechanism to call the online inference service to rapidly generate inference results.

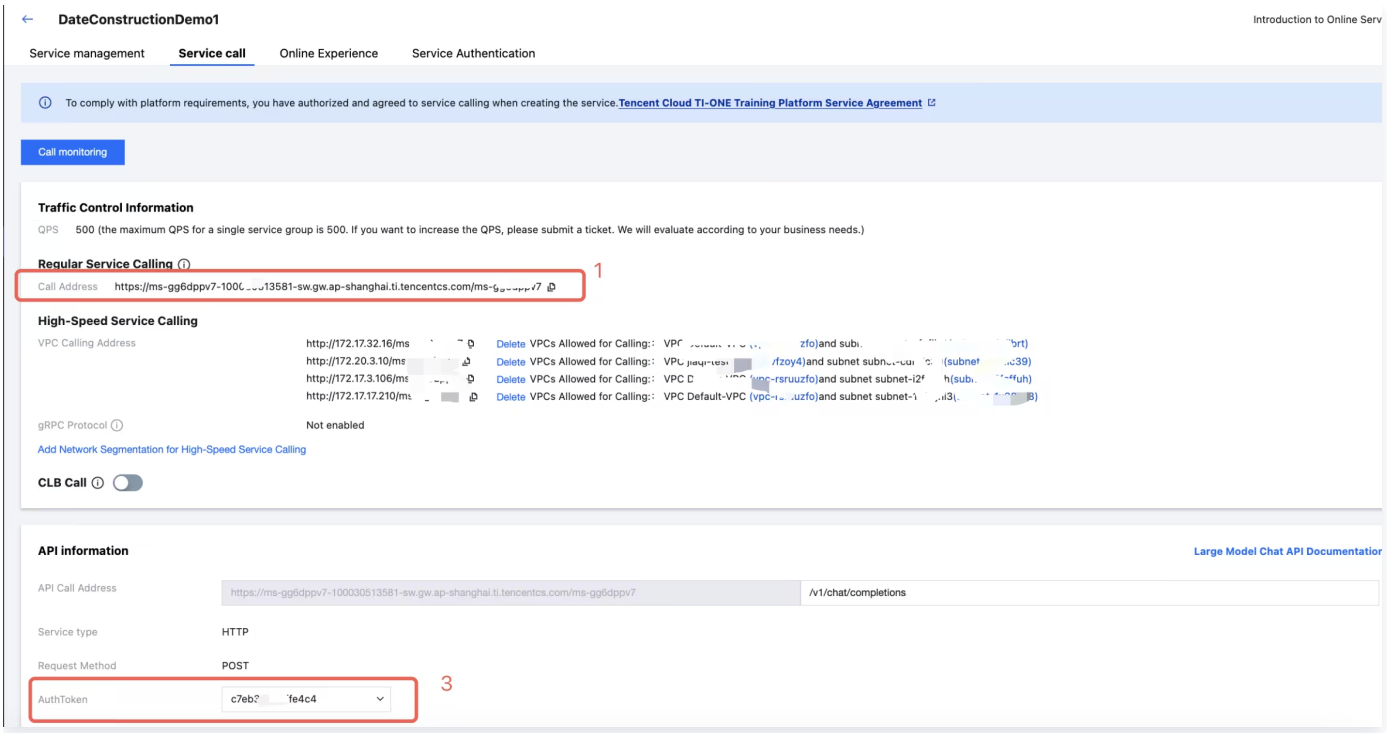


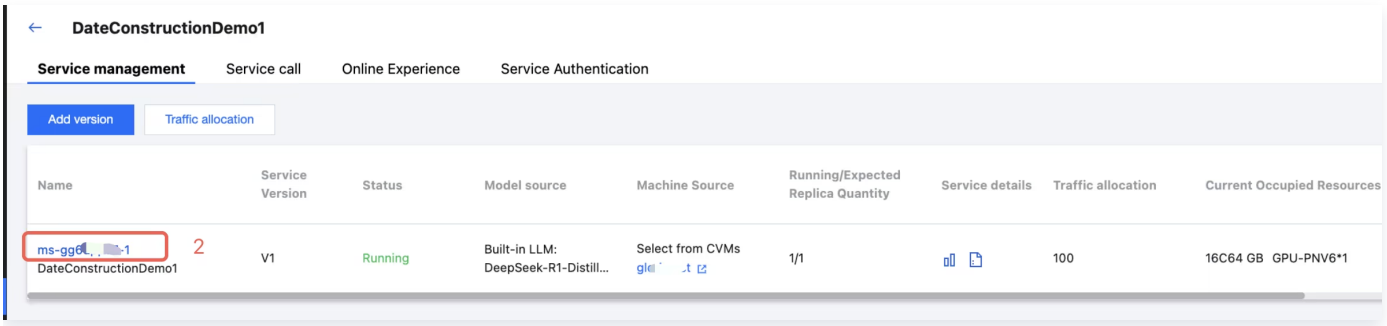
Before you run the script feature, ensure that your custom online service in the **Model Services > Online Services** module is already in the Running status. Your objective is to use the existing sample set to batch call this online service to obtain sample inference results.



Click the online service name to view details, and fill in the information marked in the following three red boxes in the parameters in the data preprocessing script respectively.

©2013-2026 Tencent Cloud. All rights reserved.






```
%reload_ext autoreload
%autoreload 2

import os
from tools.generate_reasoning_stream import invoke_service

# Specify input/output parameters
input_jsonl_file = os.path.join("raw_dataset_files", "single_qa_example.jsonl")
output_jsonl_file = os.path.join("raw_dataset_files", "single_qa_example_deepseek_generation.jsonl")
# Specify the prompt column name in input data
prompt_column = "input"
# Online service endpoint
model_addr = "https://ms-xxxxx-100012345677-sw.gw.ap-shanghai.ti.tencentcs.com/ms-xxxxx" 1
# Online service model ID
model_service_id = "ms-xxxxx" 2
# Online service token
model_token="Bearer xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx" 3
# max_concurrent specifies the maximum concurrency. An R1 service deployed on 2 machines can support up to 128 concurrent calls with high
max_concurrent = 10

# Start invocation
await invoke_service(
    input_dataset_file=input_jsonl_file,
    prompt_column=prompt_column,
    model_addr=model_addr,
    model_token=model_token,
    model_service=model_service_id,
    output_file=output_jsonl_file,
    max_concurrent=10
)
```

After filling in your online service parameters, modify the path of your data sample input_jsonl_file/output_jsonl_file, and then click to run the notebook to view the preprocessing

©2013-2026 Tencent Cloud. All rights reserved.

progress of this data distillation task in real time.

```
# Start invocation
await invoke_service(
    input_dataset_file=input_jsonl_file,
    prompt_column=prompt_column,
    model_addr=model_addr,
    model_token=model_token,
    model_service=model_service_id,
    output_file=output_jsonl_file,
    max_concurrent=10
)
```

```
Generating train split: 0 examples [00:00, ? examples/s]
Map:   0%|          | 0/50 [00:00<?, ? examples/s]
Generating responses:  56%|[redacted]     | 28/50 [00:00<00:00, 109.43row/s, active=10]
API error: Route Not Found. Please check your request uri & header, especially X-TC-Action & X-TC-Version
```

©2013-2026 Tencent Cloud. All rights reserved.

Task-based Modeling
Overview of Task-based Modeling

Last updated: 2026-02-26 15:29:31

Task-based Modeling provides a wizard-based method to submit training tasks for model building. You can directly use the built-in images of the platform to quickly submit training tasks with mainstream high-performance and distributed training frameworks. You can also start tasks through custom training images. The detailed description of its functional modules is as follows:

- Training task creation: creates and submits a training task based on a built-in framework or custom training image. You can use platform data sets, COS data, or CFS instances as training samples. You can also set algorithm parameters for the training task, enable log shipping, and bind a VPC network.
- Task operation management: provides conventional management operations, including starting, stopping, and deleting training tasks. This module also provides the task replication feature that allows you to quickly start training tasks with different configurations to compare the effects of model training.
- Task operation monitoring: provides running logs of training, and visual monitoring of training metrics, training resource consumption, and training tasks started via TiKit.
- One-click model release: provides the ability to release models to the model repository with a single click, and supports updates of model versions.
- TensorBoard: supports starting TensorBoard for task monitoring.

©2013-2026 Tencent Cloud. All rights reserved.

Creating a Task

Last updated: 2026-03-12 10:37:15

Creation Steps

Filling in the Basic Information

1. Log in to the Tencent Cloud TI-ONE Platform (TI-ONE) console, choose **Training Workshop** > **Task-based Modeling**, and click **New** to start creating a training task.
2. On the basic information configuration page, fill in the following information:
   - Task Name: Only Chinese, English, numbers, underscores (_), and hyphens (-) are supported. It must start with a Chinese character, English letter, or number.
   - Region: region where the training task is located. It defaults to the region of the current list page.
   - Training image: You can choose from built-in training images of the platform, custom images, or built-in LLMs. For the list of built-in training images, see Built-in Training Image List. For the list of built-in training images, please refer to the Built-in Training Image List. Custom images support selecting images from Tencent Container Registry or filling in an external image address. For private images, a username and password are required. For custom image specifications, see Specifications for Custom Training Images. For instructions on using built-in LLMs for training, see Fine-Tuning Built-in Open-Source LLMs.
   - Training Mode: For the training modes supported by different training frameworks, see Built-in Training Image List.
   - CVM Instance Source and Resource Request: You can choose to select from CVM instances or purchase from the TI-ONE platform. If you choose to purchase from the TI-ONE platform, you need to select the CVM instance specifications and the number of nodes. If you choose to select from CVM instances, you need to first create a resource group and purchase nodes. After selecting the resource group, select the corresponding computing resources. For the billing specifications supported by the platform, see Billing Overview.
   - Use RDMA: When configuring full GPU cards and selecting more than 1 node, you can configure whether to enable RDMA. RDMA is enabled by default for multi-node tasks configured with full 8 GPU cards per node. When RDMA is enabled, tasks are preferentially scheduled to nodes that support RDMA. When there are insufficient IP addresses for RDMA NICs on a single node, tasks may be scheduled to multiple nodes.

> **Note:**

©2013-2026 Tencent Cloud. All rights reserved.

1. Once a resource group is selected, an overview of the remaining GPUs in the resource group will be displayed. The displayed information includes the total number of GPU cards of each GPU model, the number of full GPU cards, and the number of non-full GPU cards (fragmented GPU cards). This helps you quickly understand the GPU distribution in the selected resource group. Based on the current task scenario, you can use full resources or non-full (fragmented) resources to effectively reduce overall resource fragmentation and improve the overall utilization of GPUs.
2. Click **View Details** to display a detailed resource dashboard on the right side of the current page. This dashboard displays the remaining available resources and total resources for each GPU card type. Click the drop-down icon to display all running tasks/services on the current node. This helps you quickly understand resource usage and facilitate resource usage coordination among teams.

- Description: You can add a description of up to 500 characters for remarks.
- Cloud Log Service (CLS) log shipping: CLS log shipping is disabled by default. The TI-ONE console displays logs for 15 days by default. If you want persistent storage of logs and services such as log search, you can enable CLS log shipping. After enabling this feature, you can ship logs to CLS (ensure that CLS has been activated).
- Automatic restart: You can configure an automatic restart policy for the task. You need to set the maximum number of restart attempts, up to 10. If the number of restarts exceeds this limit, the task will be marked as abnormal. The trigger condition for automatic restart is an abnormal exit during task execution. This feature currently only supports training tasks with the billing mode set to yearly/monthly subscription and in the MPI, DDP, or Horovod training mode. Event information for automatic restarts can be viewed by selecting **Events on the Task Details** page.
- Health check:
  - You can choose to enable the health check for the task. Once this feature is enabled, it will take some time to perform health checks on the nodes scheduled for the task. The check is performed before task execution, including both manual starts and automatic restarts. If the health check fails or times out, the task will be terminated, and any occupied resources will be released. Currently supported check items include **NCCL network check** and **slow node check**. The conditions for the checks are that the requested resources in the task include full GPU cards, and the training mode is DDP, MPI, or Horovod. An additional condition for the slow node check is that the number of nodes configured for the task is at least 2.
  - You can also configure the maximum check duration for the health check. When the check time exceeds this duration, and the check task has not ended normally, the training task will be terminated, and the occupied resources will be released. During

©2013-2026 Tencent Cloud. All rights reserved.

the health check, you can choose **Task Details > Logs** and set the log type to **Platform Logs (Health Check)** to view the check logs and data.

Filling in Information for Task Configuration

On the task configuration page, you need to configure the algorithm, data, and input and output information about this training task. The configuration items are described as follows:

1. Storage path settings: Supported storage types include data sets, data sources, COS, CFS (including CFS Turbo), GooseFSx, GooseFS, and EMR (HDFS). GooseFS and GooseFSx are only supported for resource groups selected from CVM instances. For each configured storage path, its purpose can be selected, including self-built models, self-built code, self-built data, training data, and others. When the storage type is set to CFS, the purpose can also be set to the built-in models of the platform, allowing direct mounting of models from TI-ONE's built-in CFS to the training container.
   - If you select the data set type, you need to first create a data set by choosing **Data Center > Data Set Management** on the platform.
   - If you select the data source type, you need to first create a data source by choosing **Platform Management > Data Source Management**. Note: The mount permissions for data sources are divided into read-only mount and read-write mount. For data sources that are intended to receive training output results, configure read-write mount permissions.
   - If you select COS, you need to select the COS path where the data is located.
   - If you select CFS, GooseFSx, GooseFS, or EMR (HDFS), you need to select the CFS instance, GooseFSx instance, GooseFS cluster, or EMR cluster from the drop-down list and fill in the data source directory that you want the platform to mount.

   For the above-mentioned data sources, you can define the mount path of the data within the training container during configuration. You need to fill in this path in your code to obtain the data. When creating a task, you can select multiple data sets or data paths, set different container mount paths respectively, and mount them to the container for the training algorithm to read.
   - Must-knows for using EMR (HDFS): The platform will access and mount HDFS using the Hadoop identity by default. To use another identity, upload the relevant configuration file in accordance with the following specifications for the code package.
     - The username and keytab file are provided by the user and placed in the code package.
     - Specifications for the code package:
       / `<emr_id>` /username.txt. (The content is the username, such as hadoop/172.0.1.5. When Kerberos authentication is not enabled, the default username hadoop can be used if the file does not exist or is empty. After Kerberos authentication is enabled, the default username becomes unavailable.)

©2013-2026 Tencent Cloud. All rights reserved.

- / `<emr_id>` /emr.keytab (The content is the keytab authentication file.)
  (Since the platform supports multiple authentications for multiple EMR instances, you need to add `<emr_id>` to the directory, with a value such as emr-1rnhggsh.)

2. Git storage: You can select a Git repository and configure a storage path within the container. When the task starts, the files in the repository will be downloaded to the specified storage path. You need to choose **Training Workshop > Git Repositories** to create a repository first.
3. Startup command: You need to fill in the program entry command, which supports multiple lines. The default working directory is /opt/ml/code.
4. Training output: Select the COS path where you need to save the training output. By default, the platform will regularly upload the data in the /opt/ml/output path to the output COS path. To release the trained model to the model repository with a single click, you need to save the model output to the /opt/ml/model path. The platform will upload the data in this path to the COS path after the training is completed. If you set training storage to the file system, such as CFS, you can also choose not to configure the training output and directly write the training output to the mounted CFS file path.
5. Tuning Parameter: The filled-in hyperparameter JSON will be saved as the /opt/ml/input/config/hyperparameters.json file. You need to parse your code.
6. SSH Connection: You can choose whether to enable SSH connection (currently only supported when the CVM instance source is set to Select from CVM Instances). After enabling the SSH connection, you can access this instance from other CVM instances. You need to fill in the content of the ~/.ssh/id_rsa.pub file on the CVM instance initiating the SSH login. If the file does not exist, you can generate it using the ssh-keygen command. When initiating an SSH login, check whether the private key matches. To initiate an SSH login from multiple CVM instances, you need to fill in multiple public keys. You can add multiple keys (press Enter to enter multiple keys).
7. Add Port: If you need to access services started in dev machines externally, you can enable the feature for custom port configuration (currently only supported when the CVM instance source is set to Select from CVM Instances):

> ⚠ **Note:** Ensure that the resource group and the Cloud Load Balancer (CLB) instance belong to the same VPC network.

- Service Name: Enter a name to distinguish between different custom services started in dev machines.
- Access Protocol: Configure an access protocol, which can be TCP or UDP.
- Listening Port: container port listened by the custom service process running in dev machines. The port is used to receive externally sent network requests. Only ports 1 to 65535 are supported.
- Method for Service Access:

©2013-2026 Tencent Cloud. All rights reserved.

- For SSH connections, access via Pod IP addresses within the VPC network is supported by default. The access address can be viewed on the details page after instance creation. You can choose to perform port mapping through the CLB access method.
- For other custom services, the CLB access method is selected by default for port mapping.

- Access Port: If you select CLB access, you need to add a mapping port. After the port is filled in, a listener will be created under the selected CLB instance, and a port will be allocated. Be careful not to fill in a port that is already occupied by a listener. Ports 1 to 65535 are supported.
- Select CLB: After selecting CLB access, you need to select a CLB instance under your account. If no instance is available under the current account, you can create an instance in the CLB console.

After the instance is created, the custom port information and access address can be viewed on the dev machine details page.

Additionally, during the process of task configuration, note that the price of your current task configuration will be displayed in real time at the bottom. Once all information is configured, the task is created.

Description of the Preset Process for Built-in LLMs

Task-based Modeling includes multiple fine-tuning templates for LLMs. You can start tasks for fine-tuning built-in LLMs with a single click. For detailed best practices, see Fine-Tuning Built-in Open-Source LLMs. The following are descriptions of built-in fields:

Preset Storage Path Settings

- For the first line Platform CFS: By default, the system has configured the supporting training code for fine-tuning the LLM for you.
- For the second line Platform CFS: By default, the system has configured a set of sample data for fine-tuning the LLM for you. To use your custom business data to fine-tune the LLM, you can delete this line and add other storage sources at the bottom.
- For the third line Platform CFS: By default, the system has configured a built-in model of the platform for you.
- For the fourth line User CFS: You need to select your CFS instance and source path here. Container Mount Path is automatically populated by the system and requires no modification. To use a different CFS instance as the training output, you can delete this line and add a new one.

> ⚠ **Note:**

©2013-2026 Tencent Cloud. All rights reserved.

> If you use your own business data for fine-tuning, you need to use the format specified by the platform or follow the description provided in the dataset_info.json data configuration file of llamafactory. For details, click here to view more information.

Preset Startup Command

The platform populates the startup command by default. Generally, you do not need to modify the startup command.

Preset Parameters for Tuning

The platform provides multiple preset parameters. You can directly modify the hyperparameter JSON to iterate the model. The following table describes the hyperparameters:

Hyperparameter	Default Value	Description
Stage	sft	Indicates the training mode, which can be sft, pt, or dpo. The default value is sft. Currently, the Stage parameter is only supported by Qwen3 series models.
Epoch	2	Number of training iterations.
BatchSize	1	Number of samples processed in each training iteration. A larger BatchSize value can speed up training, but also increases memory usage.
LearningRate	1e-5	A hyperparameter for updating weights during gradient descent. A value that is too high may prevent the model from converging, while a value that is too low will slow down the convergence speed.
Step	750	Interval (in steps) at which a model checkpoint is saved. Saving more checkpoints requires more storage space.
UseTilearn	True	Whether to enable Tencent's self-developed acceleration framework. Valid values are true and false. Setting this parameter to true will enable Tencent's self-developed acceleration framework for training. 3D parallel acceleration requires 8 or more GPU cards and necessitates configuring the pipeline parallelism (PP) and Tensor parallelism (TP) parameters. For details, see the angel-tilearn documentation. If this parameter is set to false, the open-source acceleration framework is used by default. This parameter is only available for certain models.

©2013-2026 Tencent Cloud. All rights reserved.

FinetuningType	Lora	When the Stage parameter is set to sft, you can customize the fine-tuning training mode, which can be LoRA or Full. In LoRA mode, the system keeps the parameters of the pre-trained LLM fixed and applies low-rank decomposition to the weight matrix, updating only the low-rank parameters during the training process. In Full mode, all model parameters are fully updated during fine-tuning, which requires more training resources.
MaxSequenceLength	2048	Maximum text sequence length, which can be reasonably configured based on your business data length. For example, if most business data is less than 2048 characters in length, setting MaxSequenceLength to 2048 will truncate any data exceeding this length to 2048, thereby reducing GPU memory pressure.
GradientAccumulationSteps	1	A parameter from the Hugging Face Trainer. The default value is 1, which increases the value of BatchSize.
GradientCheckPointing	True	A parameter from the Hugging Face Trainer. The default value is True, which is a time-for-GPU memory policy. Enabling this parameter optimizes GPU memory usage but slows down training speed.
DeepSpeedZeroStage	z3	DeepSpeed ZeRO stage configuration. The optional values are z0, z2, z2_offload, z3, and z3_offload. The default value is z3. This parameter is only available for certain models.
ResumeFromCheckpoint	True	Whether to automatically resume training from an existing checkpoint file. The default value is True, which indicates that if a checkpoint file exists in the output directory, training resumes from the latest checkpoint. If this parameter is set to False, the training process will restart from scratch. If this parameter is set to False and the output directory is not empty, an error will occur. It is recommended to use an empty directory for training output. To enable forced overwriting, manually add the parameter "overwrite_output_dir": true to enable file overwriting.
TilearnHybridTPSize	1	A Tilearn 3D parallelism parameter that specifies the dimension for TP, with a default value of 1. This parameter is only available for certain models.
TilearnHybridPPSiz	1	A Tilearn 3D parallelism parameter that specifies the dimension for PP, with a default value of 1. This parameter

©2013-2026 Tencent Cloud. All rights reserved.

e		is only available for certain models.

©2013-2026 Tencent Cloud. All rights reserved.

Task Management

Last updated: 2026-02-26 15:42:14

Introduction

After a task is created, this training task record will be shown on the task list page. The list displays the task name, training image, machine source, occupied resources, tag, status (submitting, queuing, starting, running, abnormal, stopping, stopped, and completed), running time, creator, training start time, update time, and monitoring and operation. Operations include Replicate, TensorBoard, Stop, Start, and Delete.

Brief Introduction of Task Status:

- Submitting: Status of a task from the moment it successfully initiates a submission request until it enters the queue.
- Queuing: Status of a task after it is successfully submitted and enters the queue. When the status is Queuing, you can click to display the queuing duration and priority of the task.
- Starting: Status of a task after it is successfully dequeued, scheduled with resources, and starts initialization.
- Running: Status of a task after it is successfully loaded and running.
- Exception: Status of a task if it exits exceptionally.
- Stopping: Status of a task if it is manually stopped by the user and its resources are repossessed.
- Stopped: Status of a task if it is manually stopped by the user and terminated.
- Completed: Status of a task if training is completed and it exits normally.

Viewing My Tasks

There is a shortcut option to view my tasks in the upper-right corner of the list page. After the shortcut option is checked, only the tasks created by the currently logged-in sub-account will be displayed.

Replicating Tasks

To compare the training effectiveness of different datasets or hyperparameter configurations across multiple tasks, you can replicate training tasks. Click **Copy Training Task** to navigate to the task creation window. Simple modifications can be made based on the original task configuration information to quickly create a new task.

Tensorboard

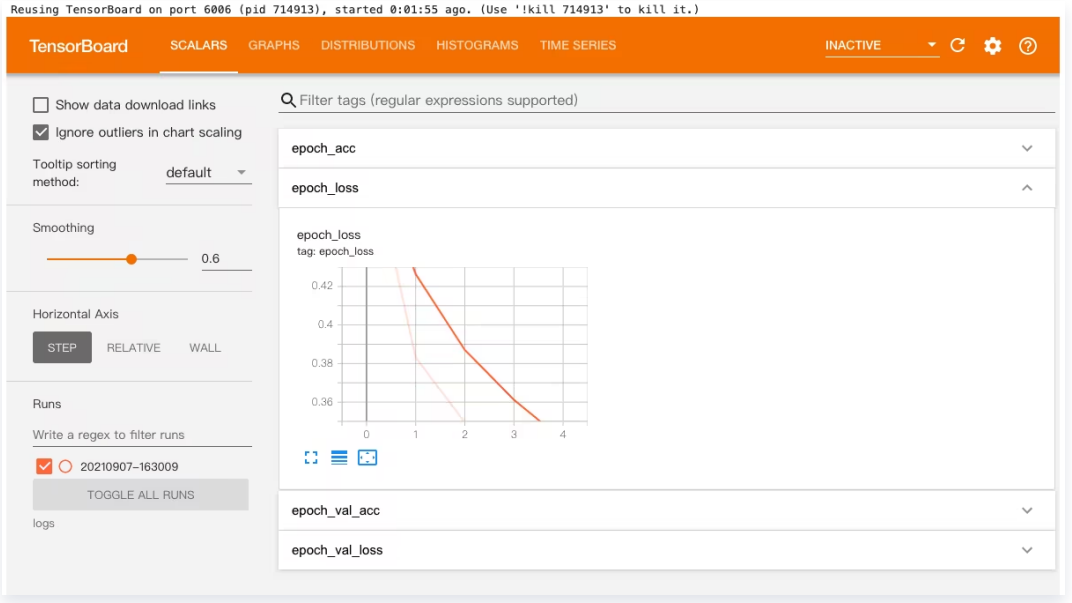
The platform supports reading summary data from Cloud File Storage (CFS) and Cloud Object Storage (COS) to generate a TensorBoard monitoring dashboard. The specific procedure is

©2013-2026 Tencent Cloud. All rights reserved.

shown as follows:

- Click **Tensorboard** in the operation column of the Task List to start the configuration of TensorBoard.
- If your training output file is stored in CFS, select CFS to store the output summary, and select the storage path as the CFS file system and source path configured for your current training task (select one path if you have configured multiple paths). Fill in the directory within the container where your summary data resides. The platform will display the local path of the container where your CFS system is mounted by default, and you only need to fill in the specific subdirectory where the summary file is located, as shown below:
- If your training output files are stored in COS, select COS to store the output summary. Note that if COS is selected, the storage path defaults to the training output COS path configured for the task, namely, + /< job_id> + /summary. The summary directory defaults to /opt/ml/summary/ (path within the container). You are not required to configure the summary directory. You need to output the summary monitoring data to /opt/ml/summary/ in advance in the training code. Otherwise, the TensorBoard panel cannot be created.
- Once the configuration is completed, click **Confirm** to enter the TensorBoard info page. Click **Click to jump** on the page to navigate to the TensorBoard panel.

  It may take 1–2 minutes to start TensorBoard. During this period, do not close the redirection page.



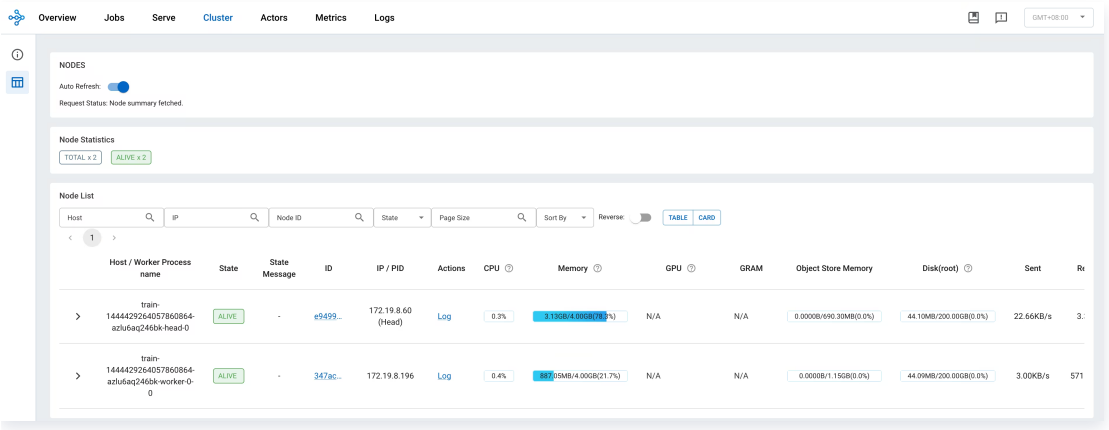
  Currently, the TensorBoard service is provided free for users. You may encounter a situation of insufficient platform resources. If you encounter an "unavailable" status in this service, please wait patiently and try again later.

Ray Dashboard

©2013–2026 Tencent Cloud. All rights reserved.

After a task in Ray training mode is selected, the platform supports opening the Ray Dashboard with one click. Users can enter the panel to view task and cluster information.

In the task list, click Dashboard:
Dashboard page:



> **Note:**
> TI-ONE's Ray Dashboard and Tensorboard features are capabilities provided by TI-ONE through adaptation and secondary development based on open-source software. For such software, TI-ONE offers comprehensive technical support services. However, TI-ONE does not provide compensation or indemnification for business losses caused by defects inherent in the open-source software itself during usage. If you encounter defects or operational issues during use, you may contact us via a support ticket.

Stopping or Starting a Task

You can stop an ongoing training task manually. A stopped task can be restarted.

Publishing to Model Repository

For the tasks that have completed training (only those tasks that write the output data to the COS storage path as required are supported), the output can be published to the model repository. Click **Publish to Model Repository**, and you can create a model or add an existing model version.
To create a model, you need to input the model name. To add a model version, you need to select a model in the model repository. The version number will be automatically generated (the default version number of the new version will be automatically incremented by 1).

Deleting a Task

You can manually delete records for completed/stopped tasks.

Task Details

©2013-2026 Tencent Cloud. All rights reserved.

Click the name of the training task to go to the task details page.

Basic Info

The basic configuration page displays the information about basic configurations, job parameters, and resource configurations of the training task. The task tags can be updated.

Instance List

The instance list page shows the instances of current training tasks. On this page, you can view the instance ID, occupied resources, and status. Additionally, you can directly access the instance container for a training instance. You can click **Enter Container** on the page to enter the webshell page, and execute command-line operations on the training instance container.

Execution Records

The execution record list page displays the execution history of the current task in chronological order, including manual executions and abnormal automatic restarts. You can quickly navigate to the corresponding log and event pages to view details of each execution record.

Monitoring

Task monitoring has two dimensions: training metrics monitoring and resource monitoring. The training metrics monitoring page shows the monitoring metrics reported by users in a customized manner. The metrics monitoring information generated during the user's training process needs to be reported by the user in the training script following the platform rules. For details, check the **push_training_metrics() command** in Tikit - Submitting Training Tasks. Resource monitoring information shows the CPU usage, memory usage, used memory, VRAM usage, used VRAM of the training task. Fine-grained GPU-level monitoring metrics will be displayed for full-GPU tasks. If you have submitted a full-machine training task, you can view the node RDMA usage monitoring information.

Code Example for Reporting Training Metrics:

```
## In the valide function, upload training metrics acc1 and acc5.

def validate(val_loader, model, criterion, epoch, args):
    batch_time = AverageMeter('Time', ':6.3f')
    losses = AverageMeter('Loss', ':.4e')
    top1 = AverageMeter('Acc@1', ':6.2f')
    top5 = AverageMeter('Acc@5', ':6.2f')
    progress = ProgressMeter(
        len(val_loader),
        [batch_time, losses, top1, top5],
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        prefix='Test: ')

    # switch to evaluate mode
    model.eval()

    with torch.no_grad():
        end = time.time()
        for i, (images, target) in enumerate(val_loader):
            if args.gpu is not None:
                images = images.cuda(args.gpu, non_blocking=True)
            target = target.cuda(args.gpu, non_blocking=True)

            # compute output
            output = model(images)
            loss = criterion(output, target)

            # measure accuracy and record loss
            acc1, acc5 = accuracy(output, target, topk=(1, 5))
            losses.update(loss.item(), images.size(0))
            top1.update(acc1[0], images.size(0))
            top5.update(acc5[0], images.size(0))

            # measure elapsed time
            batch_time.update(time.time() - end)
            end = time.time()

            if i % args.print_freq == 0:
                progress.display(i)

  ## Call Tikit commands to upload metrics.

        client.push_training_metrics(int(time.time()), {"acc1":
float(format(top1.avg, '.3f')), "acc5": float(format(top5.avg, '.3f'))},
epoch=epoch)
        # TODO: this should also be done with the ProgressMeter
        print('TIACC - * Acc@1 {top1.avg:.3f} Acc@5 {top5.avg:.3f} Epoch=
{epoch}'
              .format(top1=top1, top5=top5, epoch=epoch))

    return top1.avg
```

©2013-2026 Tencent Cloud. All rights reserved.

After task completion, you can view the metrics under "Task-based Modeling/Monitoring" (there may be a slight delay).

Training Logs

The training task page displays instance-level training logs, which can be refreshed in real time. It supports full-text search and allows viewing historical logs of the training task by selecting specific time ranges. By default, the platform stores logs for up to 7 days. For persistent storage, use CLS log shipping. You can enable or disable CLS log shipping on this page. Additionally, the platform categorizes logs for filtered viewing, including Platform Logs (Health Checks), Platform Logs (Initialization), and Training Code Logs, allowing targeted review of specific log categories.

Events

The platform supports viewing events that occur during task execution. You can diagnose issues based on these events.

©2013-2026 Tencent Cloud. All rights reserved.

User Guide on Distributed Training

Last updated: 2026-03-12 10:14:35

Task-based Modeling supports submitting distributed training tasks across multiple CVM instances and GPU cards. Tencent Cloud TI-ONE Platform (TI-ONE) supports various distributed training modes, including DDP, Ray, MPI, and PS-Worker. This document will elaborate on how to use different training methods within TI-ONE. Additionally, large-scale distributed training tasks require remote direct memory access (RDMA) technology to achieve high-throughput, low-latency network communication, thereby enhancing training efficiency. Finally, this document will also introduce how to conduct distributed training on TI-ONE using RDMA-based high-performance GPU instances.

1. Supported Distributed Training Modes and Use Instructions

TI-ONE Task-based Modeling supports various distributed training modes, including DDP, Ray, MPI, and PS-Worker. The following is an overview of the distributed training modes supported by the platform and their use cases.

Distributed Training Mode	Mode Introduction
DDP	PyTorch DistributedDataParallel (DDP) is a data-parallel distributed training method. DDP creates multiple processes for model training, using the ring-all-reduce approach for inter-process communication to complete gradient exchange and parameter updates.
MPI	MPI is a message-passing-based parallel programming technology. The platform allows you to initiate MPI-based distributed training tasks, while also supporting common MPI-based training frameworks such as Horovod and DeepSpeed.
Ray	Ray is an open-source distributed computing framework that simplifies the development and deployment of distributed machine learning applications. Ray provides a set of APIs and infrastructure that enable developers to easily scale stand-alone training code to a distributed environment. Ray supports multiple parallelization policies, including data parallelism and model parallelism. In the LLM domain, Ray is also commonly used as a distributed framework to accelerate reinforcement learning training processes.
PS-Worker	Parameter server (PS) training is a common data-parallel method used to scale model training across multiple CVM instances. The training cluster consists of workers and PSs. Parameters are stored on a PS. In each training iteration, the PS distributes parameters to

©2013-2026 Tencent Cloud. All rights reserved.

	workers. After the workers complete their calculations, they send gradients back to the PS for updates.

Use Instructions for the DDP Mode

The PyTorch Distributed Data Parallel (DDP) training mode supports data parallel training in PyTorch. The data parallel mode can process multiple data batches simultaneously across multiple processes. The input data batches of each process do not overlap. Each process calculates gradients and uses the ring-all-reduce algorithm to complete synchronization with other processes.

Usage Methods

1. When creating a training task on the **Task-based Modeling** page, set the training mode to DDP, and configure single-node resources and the number of nodes. The startup command will be executed on each node.
2. The DDP training mode includes two types of roles, Master and Worker. The role numbered 0 is Master (corresponding to RANK=0 in the environment variable), which is used to save the model.
3. TI-ONE creates a corresponding instance according to the task configuration and injects relevant environment variables, such as the instance group information contained in the task and the role of the current instance. The worker will wait for the Master to start properly and the network to be connected. The following table shows the list of environment variables that are automatically injected when Task-based Modeling starts:

   Built-in environment variables

Variable Name	Variable Description	Example
NODE_LIST	Common environment variables for training tasks: list of task nodes and information about the number of GPU cards on each node.	NODE_LIST=timaker-xxxyyy-launcher.training-job.svc.cluster.local:1,timaker-xxxyyy-worker-0.training-job.svc.cluster.local:1
INDEX	Common environment variables for training tasks: index of the current node in NODE_LIST, starting from 0.	INDEX=1
MASTER_ADDR	IP address of the master node of the DDP training task.	MASTER_ADDR=10.35.110.11

©2013-2026 Tencent Cloud. All rights reserved.

MASTER_PORT	Port of the master node of the DDP training task.	MASTER_PORT=23456
WORLD_SIZE	Number of nodes of the DDP training task.	WORLD_SIZE=2
RANK	Current node of the DDP training task.	RANK=1
GPU_NUM	Total number of GPU cards in the task.	GPU_NUM=2
GPU_NUM_PER_NODE	Number of GPU cards per node.	GPU_NUM_PER_NODE=1

4. If the exit code of any instance is non-zero during the training process, the training task will fail. If all instances succeed, the training task succeeds.

Startup Method Example

The sample command to start DDP distributed training is shown as follows:

```
python -m torch.distributed.launch --nproc_per_node $GPU_NUM_PER_NODE --
nnodes $WORLD_SIZE --node_rank $RANK --master_addr $MASTER_ADDR --
master_port $MASTER_PORT
```

The following table shows the correspondence between DDP distributed training parameters and platform environment variables:

Variable Name	Variable Description
nproc_per_node	Number of processes running on a single instance. When GPUs are used, this variable typically equals the number of GPU cards per CVM instance, which corresponds to the value of the GPU_NUM_PER_NODE environment variable.
nnodes	Corresponds to the value of the WORLD_SIZE environment variable.
node_rank	Corresponds to the value of the RANK environment variable.
master_addr	Corresponds to the value of the MASTER_ADDR environment variable.
master_port	Corresponds to the value of the MASTER_PORT environment variable.

For more DDP training parameters, see the TORCHRUN documentation on the official website.

©2013-2026 Tencent Cloud. All rights reserved.

For the built-in environment variables involved in the startup command, the platform will inject them when starting a task-based modeling task. When debugging code in Notebook or locally, developers need to assign values to the corresponding environment variables first. For convenience in debugging, you can set default values for the corresponding environment variables. Example:

```
MASTER_ADDR=${MASTER_ADDR:-localhost}
MASTER_PORT=${MASTER_PORT:-23456}
NNODES=${WORLD_SIZE:-1}
NODE_RANK=${RANK:-0}
GPU_PER_NODE=${GPU_NUM_PER_NODE:-$(nvidia-smi -L | wc -l)}
python -m torch.distributed.launch --nproc_per_node $GPU_PER_NODE --nnodes
$NNODES --node_rank $NODE_RANK --master_addr $MASTER_ADDR --master_port
$MASTER_PORT
```

Use Instructions for the MPI/Horovod Mode

MPI is a messaging standard used for distributed parallel training. The platform allows you to initiate distributed training tasks in MPI mode and supports common training frameworks based on MPI communication, such as Horovod. The Horovod training mode is more natively adapted to tasks trained with the Horovod framework. This document takes the above two training modes as examples to introduce how to initiate distributed training tasks on a machine learning platform.

Usage Methods

1. When creating a training task on the Task-based Modeling page, set the training mode to MPI or Horovod and configure single-node resources and the number of nodes.
2. Both the MPI training mode and the Horovod mode consist of two roles: Launcher and Worker. However, both roles can execute training tasks. When a task is configured with only one instance, a Launcher instance is created by default.
3. In MPI mode, the startup command will be executed on each instance. While in Horovod mode, the startup command will only be executed on the Launcher instance. The commands for the Worker instances are configured to execute sleep infinity, waiting for commands from the Launcher. The following table shows the list of environment variables injected by default when Task-based Modeling starts:

   Built-in environment variables

Variable Name	Variable Description	Example
OMPI_MCA_orte_default_hostfile	Node information file for MPI/Horovod	OMPI_MCA_orte_default_hostfile=/etc/mpi/hostfile

©2013-2026 Tencent Cloud. All rights reserved.

	training tasks.	
GPU_NUM	Total number of GPU cards in the task.	GPU_NUM=2
GPU_NUM_PER_NODE	Number of GPU cards per node.	GPU_NUM_PER_NODE=1
NODE_IP_SLOT_LIST	List of node IP addresses and their corresponding number of GPU cards included in the task (can only be used to configure the startup command).	NODE_IP_SLOT_LIST=9.0.255.56:1,9.0.255.118:1

4. TI-ONE creates a corresponding instance group based on the task configuration and injects the relevant environment variable, which provides the information of the instance group included in the task and the role of the current instance.
5. If the exit code of any instance is non-zero during the training process, the training task will fail. If all instances succeed, the training task succeeds.

Startup Method Example

The sample content of /etc/mpi/hostfile is shown as follows:

```
train-960258573108964736-7an39bddmfpc-launcher slots=1
train-960258573108964736-7an39bddmfpc-worker-0 slots=1
```

The content is divided into two columns. The first column is the domain name of the instance, and the second column is the number of processes on the instance.

The sample command to start MPI/Horovod distributed training is shown as follows:

```
# Startup in MPI mode.
mpirun --allow-run-as-root -np $GPU_NUM -H $NODE_IP_SLOT_LIST python3
train.py --data-dir /opt/ml/input/data

# Startup in Horovod mode.
horovodrun -np $GPU_NUM -H $NODE_IP_SLOT_LIST --network-interface eth0
python3 train.py --data-dir /opt/ml/input/data
```

©2013-2026 Tencent Cloud. All rights reserved.

Additionally, the DeepSpeed framework supports configuring multi-node computing resources using OpenMPI-formatted hostfiles. For practical operations on how to launch DeepSpeed distributed training with MPI, see Guide on Running DeepSpeed Distributed Training with Task-based Modeling.

Use Instructions for the PS-worker Mode

Parameter server (PS) training is a common data-parallel method used to scale model training across multiple CVM instances. The training cluster consists of workers and PSs. Parameters are stored on a PS. In each training iteration, the PS distributes parameters to workers. After the workers complete their calculations, they send gradients back to the PS for updates.

Usage Methods

1. When creating a training task on the **Task-based Modeling** page, set the training mode to PS-Worker, and configure single-node resources and the number of nodes.
2. The PS-Worker training mode provided by the platform contains two types of roles: PS and Worker. The PS role is responsible for saving and updating parameters, requiring at least 1 instance. The Worker role is responsible for executing the training, also requiring at least one instance.
3. TI-ONE creates a corresponding instance based on the task configuration and injects the relevant environment variable TF_CONFIG. This variable provides information about the instance group included in the task and the role of the current instance. By reading TF_CONFIG, the instance obtains the number and addresses of the PS and Worker nodes in the task, and learns its own role and index from the type field within the task object.

   Environment Variables
   TF_CONFIG

```
{
    "cluster": {
        "ps": [
            "train-960252492096760832-7an13ppfli80-ps-0.train-
100031385875.svc:2222",
            "train-960252492096760832-7an13ppfli80-ps-1.train-
100031385875.svc:2222"
        ],
        "worker": [
            "train-960252492096760832-7an13ppfli80-worker-0.train-
100031385875.svc:2222",
            "train-960252492096760832-7an13ppfli80-worker-1.train-
100031385875.svc:2222"
        ]
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    },
    "task": {
        "type": "ps",
        "index": 0
    },
    "environment": "cloud"
}
```

4. If the exit code of any instance is non-zero during the training process, the training task will fail. If all instances succeed, the training task succeeds.

Use Instructions for the Ray Mode

Ray is an open-source distributed computing framework that simplifies the development and deployment of distributed machine learning applications. Ray provides a set of APIs and infrastructure that enable developers to easily scale stand-alone training code to a distributed environment. Ray supports multiple parallelization policies, including data parallelism and model parallelism.

Usage Methods

1. When creating a training task on the Task-based Modeling page, set the training mode to Ray, and configure the resources for the Head node as well as the resources and quantity for each group of Worker nodes. A maximum of 5 groups of Worker nodes can be configured.
2. The Ray training mode contains two types of roles: Head and Worker. The node with an ID of 0 is the Head node (corresponding to RANK=0 in the environment variable). This node is responsible for coordinating computing resources and scheduling tasks for the entire cluster. The Worker node is responsible for executing the specific training tasks.
3. TI-ONE creates a corresponding instance based on the task configuration and injects the relevant environment variable. The following table shows the list of environment variables injected by default when Task-based Modeling starts:

Variable Name	Variable Description	Example
HEAD_ADDR	Address of the Head node of the Ray cluster.	HEAD_ADDR=train-1282671078021627392-9qu5j2n90b9c-head-0
HEAD_PORT	Port of the Head node of the Ray cluster.	HEAD_PORT=6379

©2013-2026 Tencent Cloud. All rights reserved.

RANK	Index of the current node in the cluster. RANK0 indicates the HEAD node.	RANK=0
WORLD_SIZE	Total number of Head and Worker nodes.	WORLD_SIZE=3
GPU_NUM_PER_NODE	Number of GPU cards for the Head node.	GPU_NUM_PER_NODE=8

Preparing the Startup Command

The platform provides two methods for configuring the startup command. You can directly execute ray job submit in the startup command to configure RuntimeEnv. Alternatively, simply fill in the entrypoint command and configure RuntimeEnv parameters on the console page, and the platform will concatenate the startup command to execute ray job submit.

Method 1: Submitting Tasks Using ray job submit in the Startup Command

If you do not configure RuntimeEnv parameters on the platform page, you can use the following method to fill in the startup command:

```
ray job submit --address http://127.0.0.1:8265 --runtime-env-json='{
  "env_vars": {
    "OMP_NUM_THREADS": "32",
    "TF_WARNINGS": "none"
  }
}' -- python /opt/ml/input/data/job.py
```

Method 2: Directly Filling in the entrypoint Command in the Startup Command

If you have configured RuntimeEnv parameters on the platform console page, you can use the following method to fill in the startup command:

```
cd /opt/ml/input/data; python job.py
```

Configure RuntimeEnv on the platform console page as follows:

Preparing the Script File

You only need to use ray.init() for default initialization in your code and configure the command to execute your script in the startup command. Your task will be automatically submitted to the

©2013-2026 Tencent Cloud. All rights reserved.

cluster on the Head node, without the need to specify the address of the Head node.

- Take a simple counting task as an example. Save the following code as `job.py`.

```
import ray

ray.init()

# Define the Actor class.
@ray.remote
class Counter:
    def __init__(self):
        self.value = 0

    def increment(self):
        self.value += 1
        return self.value

# Create an Actor instance.
counter = Counter.remote()

# Concurrently call the Actor method.
futures = [counter.increment.remote() for _ in range(10)]
results = ray.get(futures)  # [1, 2, 3, ..., 10]

print("Counter result:", results).
```

- Place job.py as the training task in your specified CFS or GooseFSx directory. In Task-based Modeling, select to mount it to the `/opt/ml/code` directory, and then specify the startup command as follows:

```
cd /opt/ml/code; python job.py
```

- In the training task, you can specify a node using INDEX. For example, execute the following function on the node where the INDEX = 1 condition is met:

```
@ray.remote(resources={"Rank:1": 0.001})
def f(a, b, c):
    return a + b + c
```

©2013-2026 Tencent Cloud. All rights reserved.

Note:

1. Ray is designed for scheduling of integer resources. For example, if a task declares that it requires 1 CPU, Ray ensures that it exclusively occupies a complete physical core to avoid resource competition. Therefore, it is recommended to avoid using fractional CPU cores when configuring CPU resources for a Ray resource group. For example, if you set 0.7 cores, Ray will round this down, truncating 0.7 to 0. This can result in unexpected behavior in your program, such as tasks failing to run.
2. TI-ONE supports GPU fractional scheduling, such as allocating 0.2 GPU card resources. Within the Ray cluster, a Worker will treat 0.2 GPU card as a complete GPU, which can be configured by setting nums_gpu to 1.

2. Initiating RDMA Network Acceleration Training

RDMA is a kernel-bypass communication technology that can significantly enhance communication bandwidth in multi-machine communication scenarios. This document introduces how to use the RDMA network in the Task-based Modeling module of TI-ONE.

Prerequisites

1. At least 2 RDMA-enabled high-performance GPU nodes should be included in the resource group.
2. If a submitted distributed task is configured with 2 or more nodes, and each node is configured with 8 fully-idle GPU cards, the platform will automatically configure RDMA resources for the task.
3. The built-in LLM images on the platform support common HCC high-performance GPU instance types by default. For custom images, you need to install a user-mode RDMA driver.

How to Confirm Whether RDMA Takes Effect

For multi-machine tasks running on the platform, the following logs will be generated if RDMA is enabled:

```
[0] NCCL INFO Channel 00/0 : 8[0] -> 0[0] [receive] via NET/IBext/0/GDRDMA
```

Built-in Environment Variables of the Platform

For multi-machine training scenarios using HCC instance types, TI-ONE automatically enables RDMA and comes with the following NCCL environment variables. You do not need to explicitly set these variables when using TI-ONE.

```
NCCL_IB_GID_INDEX=3
```

©2013-2026 Tencent Cloud. All rights reserved.

```
NCCL_IB_SL=3
NCCL_CHECK_DISABLE=1
NCCL_P2P_DISABLE=0
NCCL_IB_DISABLE=0
NCCL_LL_THRESHOLD=16384
NCCL_IB_CUDA_SUPPORT=1
NCCL_IB_HCA=mlx5_bond
NCCL_NET_GDR_LEVEL=2
NCCL_IB_QPS_PER_CONNECTION=4
NCCL_IB_TC=160
NCCL_PXN_DISABLE=1
NCCL_IB_TIMEOUT=24
NCCL_DEBUG=INFO
NCCL_SOCKET_IFNAME=eth0
GLOO_SOCKET_IFNAME=eth0
TCCL_TOPO_AFFINITY=4
```

For environment variables supported by NCCL, see Environment Variables.

©2013-2026 Tencent Cloud. All rights reserved.

Guide on Configuration of Task Event Alarms

Last updated: 2026-03-24 16:39:13

Overview

Important event information generated during the training task has been connected to Tencent Cloud Observability Platform (TCOP) EventBridge. You can configure event rules in the EventBridge to push alarm notifications for related events. The currently supported events are **taskScheduling, taskRunning, taskCompleted, taskExecuteFailed, taskRestarted, taskPreempted, and taskStopped.**

Operation Steps

1. Choose TCOP > Event Set, and click **the default event set in the Cloud Service Event Set section**. This event set is used to receive events, including cloud service alarms and audits. It is region-agnostic, which means that all cloud service events from any region are delivered to the Guangzhou region by default. To configure alarms, you need to bind alarm rules under this event set.

2. Click **Manage Event Rules** to go to the event rule list. You can configure information such as event matching rules and event targets here.

3. Click **Create Rule**, enter a rule name (such as TI-ONE Alarm for Training Tasks), and set the cloud service type to **TI-ONE**.

   You can select from the drop-down menu to view the event examples, which represent the event structure of cloud service products delivered to EventBridge. Subsequent event matching rules can be customized according to the above-mentioned event structure.

   Event matching rules are based on the form mode or the custom event pattern. It is recommended to use the custom event pattern. TI-ONE currently supports targeting certain instances (such as certain training tasks) or all instances (all training tasks). New training tasks are automatically added to the queue for rule alarms. Meanwhile, the platform allows for event rule matching based on cloud product tags or task creators. The following section shows partial examples:
   - **Match by event type:**

     The following rule indicates that an alarm will be triggered when the taskExecuteFailed or taskCompleted event occurs on all training tasks under the current root account.

©2013-2026 Tencent Cloud. All rights reserved.

```
{
  "source": "tione.cloud.tencent",
  "type": [
    "tione:ErrorEvent:taskExecuteFailed",
    "tione:ErrorEvent:taskCompleted"
  ]
}
```

- **Match by sub-account UIN:**

  The following rule indicates that an alarm will be triggered when any of the taskExecuteFailed, taskCompleted, taskRestarted, and taskPreempted events occurs on the training tasks created by the sub-account 010013819411.

```
{
  "source": "tione.cloud.tencent",
  "data": {
    "taskSubUin": "010013819411"
  }
}
```

- Match by tag:

  The following rule indicates that an alarm will be triggered when any of the taskExecuteFailed, taskCompleted, taskRestarted, and taskPreempted events occurs on the training tasks **that satisfy both** the **Department:Algorithm Research and Environment:Test** tags.

```
{
  "source": "tione.cloud.tencent",
  "data": {
    "tags": [
      {
        "contain":["Department:Algorithm Research;","Environment:
Test;"]
      }
    ]
  }
}
```

©2013-2026 Tencent Cloud. All rights reserved.

The following rule indicates that an alarm will be triggered when any of the taskExecuteFailed, taskCompleted, taskRestarted, and taskPreempted events occurs on training tasks with the **Department:Algorithm Research or Environment:Test** tag.

```
{
  "source": "tione.cloud.tencent",
  "data": {
    "tags": [
      {
        "contain":["Department:Algorithm Research;"]
      },
      {
        "contain":["Environment:Test;"]
      }
    ]
  }
}
```

All the above supported types can be combined for matching. For example, the following rule indicates that an alarm will be triggered when the **taskExecuteFailed event** occurs on the training tasks **that satisfy both** the **Department:Algorithm Research and Environment:Test tags and have a creator sub-account of 010013819411.**

```
{
  "source": "tione.cloud.tencent",
  "type": [
    "tione:ErrorEvent:taskExecuteFailed"
  ],
  "data": {
    "tags": [
      {
        "contain":["Department:Algorithm Research;","Environment:
Test;"]
      }
    ],
    "taskSubUin": "010013819411"
  }
}
```

©2013-2026 Tencent Cloud. All rights reserved.

4. After completing event matching, click Next to start configuring the event target (that is, the alarm channel). Set the trigger mode to **Message Push** and the notification method to **Channel Push**. Select one or more accounts as the alarm recipients. Select **Receiving Channel**, which can be Email, SMS, Phone, or Message Center. Click Finish to return to the event rule page.

5. Once the above configurations are completed, you will receive an event notification.

©2013-2026 Tencent Cloud. All rights reserved.

Data Format Specifications for Training Built-in LLMs

Last updated: 2026-02-26 15:30:57

Overview

The built-in LLM fine-tuning module of Task-based Modeling uses the platform's training templates. You only need to prepare training data sets for fine-tuning to initiate training with a single click. This document aims to provide a detailed explanation of the data format requirements for text generation models under both supervised fine-tuning (SFT) and pre-training methods. Additionally, it outlines the data format requirements for fine-tuning embedding models for search tasks and semantic text similarity tasks.

> **Note:**
> The platform supports data set files in JSONL format (where each line in the JSONL file represents a group of sample data). Only UTF-8 is supported for encoding, and multiple files in the training data path will be merged for training.

Data Format of the Text Generation Model

SFT Data

The messages field in data samples describes multi-role chats and interactions. In multi-turn chats, the chat conversation must end with a turn from the assistant role. The following table describes the available roles:

Role	Required	Description	Field
system	No	Represents the system role setting.	content: role setting information
user	Yes	Represents the user input.	content: content of the user input
assistant	Yes	Represents the model's response.	content: content of the model's response

- Data set sample example for single-turn chats

©2013-2026 Tencent Cloud. All rights reserved.

```
{"messages": [{"role": "system", "content": "You are the TI-ONE
intelligent Q&A chatbot, capable of understanding user questions and
providing fluent answers"}, {"role": "user", "content": "Read the
following text and summarize it in one sentence.\nOn Friday, a federal
judge ruled in favor of a former UCLA basketball star who sued to end
the NCAA's control over college athletes' names, images, and likeness
rights."}, {"role": "assistant", "content": "The judge ruled in favor of
a former UCLA basketball star who challenged the NCAA's control."}]}
```

- Data set sample example for multi-turn chats

```
{"messages": [{"role": "system", "content": "You are the TI-ONE
intelligent Q&A chatbot, capable of understanding user questions and
providing fluent answers"}, {"role": "user", "content": "Read the
following text and summarize it in one sentence.\nOn Friday, a federal
judge ruled in favor of a former UCLA basketball star who sued to end
the NCAA's control over college athletes' names, images, and likeness
rights."}, {"role": "assistant", "content": "The judge ruled in favor of
a former UCLA basketball star who challenged the NCAA's control."},
{"role": "user", "content": "Read the following text and summarize it in
one sentence.\nPresident Obama apologized on Thursday to Americans who
had their insurance plans canceled because of the federal health law he
supported, even though he had repeatedly said they could keep their
plans if they liked.\nHe said in an exclusive interview with NBC News:
"I am sorry that they are finding themselves in this situation based on
my assurances.""}, {"role": "assistant", "content": "Obama expressed
regret during an interview with NBC."}]}
```

Data Sets Containing the Chain of Thought

The messages field in data samples describes multi-role chats and interactions. In responses where the role is assistant, the thought content is enclosed between <think> and </think>. In multi-turn chats, the chat conversation must end with a turn from the assistant role. The following table describes the available roles:

Role	Required	Description	Field
system	No	Represents the system role setting.	content: role setting information

©2013-2026 Tencent Cloud. All rights reserved.

user	Yes	Represents the user input.	content: content of the user input
assistant	Yes	Represents the model's response, with the thought content enclosed between <think> and </think>.	content: content of the model's response

- Sample example

```
{"messages": [{"role": "system", "content": "You are the TI-ONE
intelligent Q&A chatbot, capable of understanding user questions and
providing fluent answers"}, {"role": "user", "content": "Read the
following text and summarize it in one sentence.\nOn Friday, a federal
judge ruled in favor of a former UCLA basketball star who sued to end
the NCAA's control over college athletes' names, images, and likeness
rights."}, {"role": "assistant", "content": "<think>The user asked me to
read the above text and summarize it in one sentence. I need to identify
the subject, object, and predicate of the event, find appropriate
modifiers to qualify them, and connect them to form a summary sentence.
</think>The judge ruled in favor of a former UCLA basketball star who
challenged the NCAA's control."}]}
```

Note that the chain-of-thought format requirements for Hunyuan series models are slightly different, as described below:
The messages field in data samples describes multi-role chats and interactions. In responses where the role is assistant, the thought content is enclosed between <think> and </think>. In multi-turn chats, the chat conversation must end with a turn from the assistant. The last turn of chat is enclosed between <answer>\n and \n</answer>, with a line break \n separating the thought content and the response content. The following table describes the available roles:

Role	Required	Description	Field
system	No	Represents the system role setting.	content: role setting information
user	Yes	Represents the user input.	content: content of the user input
assistant	Yes	Represents the model's response. The last round of response has the thought content enclosed between <think>\n and \n</think>,	content: content of the model's response

©2013-2026 Tencent Cloud. All rights reserved.

		and the response content enclosed between <answer>\n and \n</answer>.	

- Sample example

```
{"messages": [{"role": "system", "content": "You are the TI-ONE
intelligent Q&A chatbot, capable of understanding user questions and
providing fluent answers"}, {"role": "user", "content": "Read the
following text and summarize it in one sentence.\nOn Friday, a federal
judge ruled in favor of a former UCLA basketball star who sued to end
the NCAA's control over college athletes' names, images, and likeness
rights."}, {"role": "assistant", "content": "<think>\nThe user asked me
to read the above text and summarize it in one sentence. I need to
identify the subject, object, and predicate of the event, find
appropriate modifiers to qualify them, and connect them to form a
summary sentence.\n</think>\n<answer>\nThe judge ruled in favor of a
former UCLA basketball star who challenged the NCAA's
control.\n</answer>"}]}
```

Data Sets Containing Tool Calls

The messages field in data samples describes multi-role chats and interactions. The tools snippet uses strings in JSON format to specify the tools that the model can call. Note that the tool calling data set should be used for models that support tool calling, such as Qwen3. In multi-turn chats, the chat conversation must end with a turn from the assistant or tool_call role. The following table describes the available roles:

Role	Required	Description	Field
system	No	Represents the system role setting.	content: role setting information
user	Yes	Represents the user input, which should appear at odd-numbered positions (starting from 1) except for the system role setting.	content: content of the user input
assistant	Yes	Represents the model's response, which should appear at even-numbered positions (starting from 1) except for the system role setting.	content: content of the model's response

©2013-2026 Tencent Cloud. All rights reserved.

tool_call	Yes	Represents tool calling, which should appear at even-numbered positions (starting from 1) except for the system role setting.	content: name and parameters of the called tool.
tool	Yes	Represents the tool calling result, which should appear at odd-numbered positions (starting from 1) except for the system role setting.	content: returned calling results

- Sample example ending with a turn from the assistant role:

```
{"messages": [{"role": "system", "content": "You are the TI-ONE
intelligent Q&A chatbot, capable of understanding user questions and
providing fluent answers"}, {"role": "user", "content": "Please answer
the following question: Xiao Li is 25 years old this year, and Xiao Wang
is 20 years old. What is the sum of their ages?"}, {"role": "tool_call",
"content": "{\"name\":\"calculate_age_sum\", \"arguments\": {\"age_1\":
25, \"age_2\": 20}}"}, {"role": "tool", "content":
"/no_think{\"age_sum\": 45}"}, {"role": "assistant", "content": "
<think>\n\n</think>\n<answer>\nAccording to my calculation, the sum of
Xiao Li's and Xiao Wang's ages is 45.\n</answer>"}], "tools": "
[{\"name\": \"calculate_age_sum\", \"description\": \"Calculates and
returns the sum of the two persons' ages.\", \"parameters\": {\"type\":
\"object\", \"properties\": {\"age_1\": {\"type\": \"number\",
\"description\": \"The first person's age.\"}, \"age_2\": {\"type\":
\"number\", \"description\": \"The second person's age.\"}},
\"required\": [\"age_1\", \"age_2\"]}}]"}
```

- Sample example ending with a turn from the tool_call role:

```
{"messages": [{"role": "system", "content": "You are the TI-ONE
intelligent Q&A chatbot, capable of understanding user questions and
providing fluent answers"}, {"role": "user", "content": "/no_think
Please answer the following question: Xiao Li is 25 years old this year,
and Xiao Wang is 20 years old. What is the sum of their ages"}, {"role":
"tool_call", "content": "
<think>\n\n</think>\n<answer>\n{\"name\":\"calculate_age_sum\",
\"arguments\": {\"age_1\": 25, \"age_2\": 20}}\n</answer>"}], "tools": "
[{\"name\": \"calculate_age_sum\", \"description\": \"Calculates and
returns the sum of the two persons' ages.\", \"parameters\": {\"type\":
```

©2013-2026 Tencent Cloud. All rights reserved.

```
\"object\", \"properties\": {\"age_1\": {\"type\": \"number\",
\"description\": \"The first person's age.\"}, \"age_2\": {\"type\":
\"number\", \"description\": \"The second person's age.\"}},
\"required\": [\"age_1\", \"age_2\"]}}]"}
```

- Description of tool calling content:
  - Tool description (tools): (expanded result)

```
"tools": "[
  {
      \"name\": \"calculate_age_sum\",
      \"description\": \"Calculate and return the sum of the two
persons' ages.\",
      \"parameters\": {
          \"type\": \"object\",
          \"properties\": {
              \"age_1\": {
                  \"type\": \"number\",
                  \"description\": \"The first person's age.\"
              },
              \"age_2\": {
                  \"type\": \"number\",
                  \"description\": \"The second person's age.\"
              }
          },
          \"required\": [\"age_1\", \"age_2\"]
      }
  }
]"
```

  - Tool calling with the Role=tool_call condition in the chat: (expanded result)

```
{
  "role": "tool_call",
  "content": "{
      \"name\":\"calculate_age_sum\",
      \"arguments\": {
          \"age_1\": 25,
          \"age_2\": 20
      }
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    }"
}
```

- Result of the tool calling with the Role=tool condition in the chat: (expanded result)

```
{
  "role": "tool",
  "content": "{
      \"age_sum\": 45
  }"
}
```

Pre-training (PT) Data

Untagged text is described using the text field in data samples.

- Sample example

```
{"text": "The layout of Luzhen's taverns is different from elsewhere:
each has a large L-shaped counter facing the street, with hot water
ready inside to warm wine at any time. Laborers who finish work around
noon or dusk would spend four copper coins—as was the case twenty years
ago, though now it costs ten—to buy a bowl of wine. They'd stand by the
counter, warming themselves through with a drink. For one more coin,
they could get a dish of salted bamboo shoots or aniseed beans to go
with the wine. If they spend ten or more coins, they can get a meat
dish. However, most of these short-coated customers are from the lower
class and generally cannot afford such luxury. Only those wearing long
gowns wander into the adjacent room of the restaurant to order wine and
dishes, sitting down leisurely to enjoy their drinks."}
{"text": "Since the age of twelve, I've worked as an attendant at
Xianheng Tavern at the entrance of our town. The proprietor said I
looked too simple-minded to serve long-gowned customers, so I was
assigned to work outside. The short-coated customers out there were
easier to deal with, though quite a few were tediously particular. They
would insist on seeing the yellow wine ladled from the jug, checking for
water at the bottom of the pot, and watching as the pot was placed in
hot water—only then would they be satisfied. Under such strict scrutiny,
diluting the wine became nearly impossible. After a few days, the
proprietor decided I couldn't handle this job either. Fortunately, out
```

©2013-2026 Tencent Cloud. All rights reserved.

```
of respect for my recommender, I couldn't be dismissed and instead was
assigned to the rather dull job of warming the wine."}
```

Custom Formats

In addition to placing data files that conform to the format described above in the training data path, you can also prepare dataset_info.json according to the LLaMA-Factory data set format to specify the training data sets. When dataset_info.json already exists in your training data path, the platform will and only will **directly read all data sets specified in this description file.** Example:

- Training data directory (such as /opt/ml/input/data/train)

```
/opt/ml/input/data/train
├── dataset1.json
└── dataset_info.json
```

- Training data dataset1.json

```
[
    {
        "messages": [
            {
                "role": "system",
                "content": "You are the TI-ONE intelligent Q&A chatbot,
capable of understanding user questions and providing fluent answers"
            },
            {
                "role": "user",
                "content": "Read the following text and summarize it in
one sentence.\nText: (CNN) - On Friday, a federal judge ruled in favor
of a former UCLA basketball star who sued to end the NCAA's control over
college athletes' names, images, and likeness rights."
            },
            {
                "role": "assistant",
                "content": "The judge ruled in favor of a former UCLA
basketball star who challenged the NCAA's control over college athletes'
names, images, and likeness rights."
            }
        ]
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    },
    {
        "messages": [
            {
                "role": "system",
                "content": "You are the TI-ONE intelligent Q&A chatbot,
capable of understanding user questions and providing fluent answers"
            },
            {
                "role": "user",
                "content": "Read the following text and summarize it in
one sentence.\nText: President Obama apologized on Thursday to Americans
who had their insurance plans canceled because of the federal health law
he supported, even though he had repeatedly said they could keep their
plans if they liked."
            },
            {
                "role": "assistant",
                "content": "During an interview with NBC, Obama
expressed regret for the cancellation of insurance plans."
            }
        ]
    }
]
```

- Data set description file dataset_info.json

```
{
    "dataset1": {
        "file_name": "dataset1.json",
        "formatting": "sharegpt",
        "columns": {
            "messages": "messages",
        },
        "tags": {
            "role_tag": "role",
            "content_tag": "content",
            "user_tag": "user",
            "assistant_tag": "assistant"
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            }
        },
    }
```

Data Formats for Embedding Models

Data for Search (IR) Tasks

Data format description:

- task_type: indicates the task type, which is fixed at ir.
- query: indicates the question to query, structured as an array composed of task-level instructions, example-level instructions, and the question text.
- positives: indicates a list of positive sample documents. There can be multiple positive samples, with each sample structured as an array composed of task-level instructions, example-level instructions, and the document text.
- negatives: indicates a list of negative sample documents. There can be multiple negative samples, with each sample structured as an array composed of task-level instructions, example-level instructions, and the document text.
- positive_scores: currently not in use, fixed as an empty array.
- negative_scores: currently not in use, fixed as an empty array.

Sample example:

```
{"task_type": "ir", "query": ["Generate representations for users' daily
queries to match web content", "", "[Question text]"], "positives":
[["Generate representations for web content text", "", "[Search web
content related to the question.]"]], "negatives": [["Generate
representations for web content text", "", "[Search Web 1 content
unrelated to the question.]"], ["Generate representations for web content
text", "", "[Search Web 2 content unrelated to the question.]"]],
"positive_scores": [], "negative_scores": []}
```

Data for Semantic Textual Similarity (STS) Tasks

Data format description:

- task_type: indicates the task type, which is fixed as sts.
- query: indicates Sentence 1 for semantic similarity calculation. It is structured as an array composed of task-level instructions,example-level instructions, and the sentence text.

©2013-2026 Tencent Cloud. All rights reserved.

- positives: indicates Sentence 2 for semantic similarity calculation, with a fixed length of 1. The sentence is structured as an array composed of task-level instructions, example-level instructions, and the sentence text.
- negatives: currently not in use, fixed as an empty array.
- positive_scores: indicates the list of scores of semantic similarity between two sentences, with a fixed length of 1.
- negative_scores: currently not in use, fixed as an empty array.

Sample example:

```
{"task_type": "sts", "query": ["Generate representations for general-
domain sentences", "", "Why is the sky blue?"], "positives": [["Generate
representations for general-domain sentences", "", "Why is the sky
blue?"]], "negatives": [], "positive_scores": [1.0], "negative_scores":
[]}
{"task_type": "sts", "query": ["Generate representations for general-
domain sentences", "", "What kind of yogurt is good for pregnant women"],
"positives": [["Generate representations for general-domain sentences",
"", "How much formula should newborns drink?"]], "negatives": [],
"positive_scores": [0.0], "negative_scores": []}
```

The task type for embedding training must be the same for all data in each JSONL file, and multiple JSONL files can be trained together.

©2013-2026 Tencent Cloud. All rights reserved.

Dev Machines
Introduction to Dev Machines

Last updated: 2026-02-26 15:31:37

Overview

> **Note:** Pay attention that the original Notebook module was upgraded to Dev Machines on February 25, 2025.

Dev Machines is a flexible online development tool tailored for developers by Tencent Cloud TI-ONE Platform (TI-ONE). It supports online coding through multiple IDEs, such as Jupyter Notebook or VS Code, and also supports remote development by connecting to dev machines over SSH. You can complete data preparation, data preprocessing, algorithm debugging, and model training in dev machines. In addition, multiple built-in mainstream development environments are provided, allowing you to easily create applications that are readily available out-of-the-box with a single click.

Core Features

- Provides a variety of resource specifications. This allows you to flexibly switch among various resources to reduce usage costs.
- Comes with multiple WebIDEs and supports remote development over SSH.
- Comes with Tikit, allowing you to submit training tasks to TI-ONE from Dev Machines.
- Provides multiple built-in images, allowing you to customize the installation of third-party libraries, export images, and save custom environments with a single click.
- Supports lifecycle scripts, allowing you to run preset shell scripts when creating or starting dev machines.
- Supports integration with Git repositories, facilitating collaborative development and versioning.

©2013-2026 Tencent Cloud. All rights reserved.

Creating a Dev Machine

Last updated: 2026-03-12 10:36:22

Overview

This document describes the operation steps to create a dev machine in Tencent Cloud TI-ONE Platform (TI-ONE). First, log in to the TI-ONE console, and click **Dev Machines** in the **menu bar** to redirect to the dev machine list page. This page displays all user-created dev machines. On the dev machine list page, click **Create New** in the upper-left corner to redirect to the setup page for creating a dev machine. The following Operation Steps section explains how to configure a dev machine.

Operation Steps

Configuring Basic Information

- **Name**: name of the dev machine. It cannot exceed 60 characters in length and can only contain Chinese characters, English letters, digits, underscores (_), and hyphens (-). It must start with a Chinese character, English letter, or digit.
- **Region**: This field cannot be modified and will automatically display the region selected on the platform.
- **Image:** You can select the image to start the dev machine. For the list of supported images, see Built-in Training Image List. Among them, the tilearn-llm training images come with the latest version of the Angel acceleration component, tilearn-llm, which can be directly used to accelerate training of certain open-source LLMs. For the detailed user guide, see Angel Training Acceleration Feature Introduction.
- **CVM Instance Source:** You can choose to select from CVM instances or purchase from the TI-ONE platform. If you choose to purchase from the TI-ONE platform, you need to select the CVM instance specifications and the number of nodes. If you choose to select from CVM instances, you need to first create a resource group and purchase nodes. After selecting the resource group, select the corresponding computing resources. For the billing specifications supported by the platform, see Billing Overview.
- **Tag**: You can add multiple tags to the dev machine.

Configuring the Image and Storage Path

- **Image:** You can select the image to start the dev machine. For the list of supported images, see Built-in Training Image List. Among them, the tilearn-llm training images come with the latest version of the Angel acceleration component, tilearn-llm, which can be directly used to accelerate training of certain open-source LLMs. For the detailed user guide, see Angel Training Acceleration Feature Introduction.

©2013-2026 Tencent Cloud. All rights reserved.

- **Storage path settings:** The data set, Premium Disk, Cloud SSD, CFS instance (including CFS Turbo), GooseFSx, or GooseFS storage type can be selected. Multiple storage paths can be configured. To ensure that the dev machine functions properly, the first selected storage path will be mounted to the default working directory of the dev machine at /home/tione/notebook.
  - If the data set type is selected, select a data set that is registered in the Data Center from the drop-down list and configure the container mount path. Note: The data set cannot be used as the first storage path setting, which means that it cannot serve as the default working directory of the dev machine.
  - If the data source type is selected, select a data source created in Platform Management > Data Source Management from the drop-down list, and configure the source path (a subpath under the storage directory of the data source) and the container mount path. Note: Dev machines can only mount data sources with read-only permissions.
  - If the Premium Disk or Cloud SSD type is selected, the platform will mount the requested hard disk to the working directory of your dev machine container. The minimum size of the storage volume for the requested hard disk is 10 GB and should be a multiple of 10. Cloud disks are only available for pay-as-you-go dev machines.
  - If the CFS instance type is selected, select a CFS instance, and fill in the source directory and container mount path of the CFS instance. The platform will mount the source directory of the CFS instance to the specified container mount path.
  - If the resource group type is Select from CVM Instances, GooseFSx, or GooseFS can also be selected. If GooseFSx is selected, select a GooseFSx instance, and configure the GooseFSx directory and the container mount path. If GooseFS is selected, select a GooseFS cluster, namespace, and COS subpath, and configure the container mount path.

Configuring Advanced Settings

Advanced settings are collapsed by default. When advanced settings are expanded, the following fields can be configured:

- **Environment Variable:** Multiple environment variables can be added.
- **CLS:** You can choose whether to activate Cloud Log Service (CLS).
- **Lifecycle Configuration:** Choose whether to use lifecycle scripts.
- **Git Storage:** This field is optional. You can choose **Git Repository** > **Add Repository** to make configurations.
- **Automatic Stop:** After this option is enabled, the instance will automatically stop when its running duration exceeds the duration you select. In the Stopped status,computing resources are no longer billed, but storage resources will continue to incur charges. Be aware of the ongoing fees. The auto-stop time is measured in hours and ranges from 1 to 24 hours.

©2013-2026 Tencent Cloud. All rights reserved.

- **SSH Connection**: You can choose whether to enable SSH connection. After enabling the SSH connection, you can access this instance from other CVM instances. You need to fill in the content of the ~/.ssh/id_rsa.pub file on the CVM instance initiating the SSH login. If the file does not exist, you can generate it using the ssh-keygen command. When initiating an SSH login, check whether the private key matches. To initiate an SSH login from multiple CVM instances, you need to fill in multiple public keys. You can add multiple keys (press Enter to enter multiple keys).
- **Add Port:** If you need to access services started in dev machines externally, you can enable the feature for custom port configuration (currently only supported when the CVM instance source is set to Select from CVM Instances):

> ⚠ **Note:** Ensure that the resource group and the Cloud Load Balancer (CLB) instance belong to the same VPC network.

- Add a port:
  - Service Name: Enter a name to distinguish between different custom services started in dev machines.
  - Access Protocol: Configure an access protocol, which can be TCP or UDP.
  - Listening Port: container port listened by the custom service process running in dev machines. The port is used to receive externally sent network requests. Only ports 1 to 65535 are supported.
  - Method for Service Access:
    - For SSH connections, shared public network access (this access address shares a public IP address and can be viewed on the details page after instance creation) and access via Pod IP addresses within the VPC network (the access address can be viewed on the details page after instance creation) are supported by default. You can choose to perform port mapping through the CLB access method.
    - For other custom services, the CLB access method is selected by default for port mapping.
  - Access Port: If you select CLB access, you need to add a mapping port. After the port is filled in, a listener will be created under the selected CLB instance, and a port will be allocated. Be careful not to fill in a port that is already occupied by a listener. Ports 1 to 65535 are supported.
- Select CLB: After selecting CLB access, you need to select a CLB instance under your account. If no instance is available under the current account, you can create an instance in the CLB console.

  After the instance is created, the custom port information and access address can be viewed on the dev machine details page.

©2013-2026 Tencent Cloud. All rights reserved.

- **VPC Network** and **Subnet** : The platform supports configuring a VPC network and subnet for the dev machine container. Once a VPC network and subnet are selected, the dev machine container will be able to access other cloud products under the same network configuration.
- **Direct Internet Access**: By default, a dev machine does not use VPC, which means that no VPC network is configured for the dev machine container. In this case, public network access is enabled by default. If a VPC network and subnet are selected, your dev machine container will be configured with the VPC network. After the configuration, the dev machine container can access other cloud services within the same VPC network. After selecting a VPC network and subnet, you can choose to disable **Direct Internet Access**. In this case, the dev machine container cannot access the public network. In the yearly/monthly subscription billing mode, this option is not required as dev machines can access the public network by default.

After the instance is created, a new instance record will be added to the dev machine list. When the instance status changes from **Starting** to **Running**, click **Open** to access the dev machine.

©2013-2026 Tencent Cloud. All rights reserved.

Managing Dev Machines

Last updated: 2026-02-26 15:33:54

Scenarios

This document demonstrates how to perform instance management, resource management, and data management on dev machines in Tencent Cloud TI-ONE Platform (TI-ONE).

Instance Management

After you **add an instance**, a new instance record will be added to the dev machine list. On this page, you can view the instance name, billing mode, used resources, tags, status, running duration, last editing time, monitoring metrics, and logs. You can also perform operations such as opening, stopping, editing, copying, and deleting dev machines.

- Click the **Monitoring and Logs** tab to view the resource monitoring status and log details of the dev machine.
  - Monitoring metrics include CPU utilization, memory utilization, GPU utilization, GPU memory utilization, and system disk utilization. When the node on which the instance resides has a local data disk, system disk utilization refers to the utilization of system disk space allocated to the Pod where the current dev machine is located. When the node on which the instance resides has no local data disk, system disk utilization refers to the utilization of system disk space allocated to the node used by the dev machine.
- Click **Open** to access the dev machine. Currently, the Jupyter Notebook or VS Code IDE can be used to open dev machines for online coding.
- Click **Stop** to stop the dev machine. After the instance is stopped, **the data in the non-mounted persistent storage path will be cleared**. To prevent data loss, move the data to a persistent storage path or save an image before the instance is stopped.
- Click **Edit** to edit certain configuration information.
- Click **Delete** to remove the instance record.
- Click **Copy** to copy the instance configuration and quickly create a dev machine.
- Click **Save Image** to save the dev machine as an image, which can be saved to both the Tencent Container Registry (TCR) Individual Edition and Enterprise Edition. Note that saved images will not include files and data from mounted external storage devices such as CBS, CFS, GooseFSx, and GooseFS paths. When saving an image, you can specify advanced settings such as EntryPoint (the default startup command or script when the image runs) and WorkDir (the default working directory when the image runs). Each image save operation generates an image build record, which can be viewed on the image build record page by clicking the dev machine name.
- Click **Remote Connection** to view the remote access address of the dev machine.

©2013-2026 Tencent Cloud. All rights reserved.

- **Automatic stop** can be enabled for the instance. After this option is enabled, the instance will automatically stop when its running duration exceeds the duration you select. In the Stopped status,computing resources are no longer billed, but storage resources will continue to incur charges. Be aware of the ongoing fees. The automatic stop time ranges from 1 to 24 hours, and should be an integer. You can set the automatic stop time when adding an instance. In the instance list, you can click the **little clock** icon next to the running duration to enable, disable, or reset the automatic stop time.

Resource Management

TI-ONE Dev Machines supports switching between resource specifications in postpaid mode. You can adjust resources based on requirements to reduce usage costs. The operation steps are as follows:

1. On the dev machine list page, select the instance for which you want to switch resources, and click **Stop** to release the current computing resources.
2. After the instance is stopped, click **Edit** to go to the settings page, where you can re-select resources, and then click **Confirm**.
3. After resources are updated, click **Start** to restart the dev machine.

> **Note**
>
> 1. A pay-as-you-go instance is billed based on the selected resource configuration and usage duration. After resources are switched, the resources before the switch are no longer billed. The platform generates fees based on the resource configuration and usage duration after the update.
> 2. After the instance is stopped or restarted, **the data in the non-mounted persistent storage path will be cleared**. To prevent data loss, move the data to a persistent storage path or save an image before the instance is stopped or restarted.
>
>   For example, if your dev machine is configured with `/home/tione/notebook` as the container mount path, after the instance is stopped or restarted, only the data in the `/home/tione/notebook` path will be persistent. The data in other paths will be cleared.

Cloud Disk Data Management

In the pay-as-you-go billing mode, if you select Premium Disk or Cloud SSD, you can select the size of the storage volume during creation. For running instances, you can adjust the storage size based on requirements. The operation steps are as follows:

1. On the dev machine list page, select the instance for which you want to adjust the storage size, and click **Stop**.
2. After the instance is stopped, click **Edit** to go to the settings page, where you can increase the storage size, and then click **Confirm**.
3. After resources are updated, click **Start** to restart the dev machine.

©2013-2026 Tencent Cloud. All rights reserved.

> **Note**
> You can increase the storage size of a hard disk, but cannot decrease it. To decrease the storage size of the hard disk, create a dev machine with the desired storage size.
> When a dev machine stops running, the cloud disks continue to incur a storage fee. The billing of the storage fee stops once the instance is deleted.

Local Disk Data Management

In the yearly/monthly subscription billing mode, for certain instance types with local disks (mostly bare metal instance types), you can specify node scheduling on the page when creating a dev machine. After scheduling is performed, the local disks of the node will be mounted to the container as a storage directory (you can still customize the mount path within the container). After the instance is stopped, the data on the local disks of the node will not be cleared. You need to manually clear the data in the dev machine. In addition, local disk data is shared among different dev machines.

©2013-2026 Tencent Cloud. All rights reserved.

Guide on Configuring an Idle Recycling Policy

Last updated: 2026-03-12 10:43:46

Background Introduction

To improve resource utilization, the platform allows you to configure a policy for automatically recycling idle dev machines. This means that an administrator can set a policy for recycling idle dev machines in the current region. The idle resources will be forcibly recycled to release computing resources when the resource utilization does not meet requirements.

Configuration Guide

Go to the **Idle Recycling Policy** tab to configure and view idle recycling policies. To perform these operations, the account must have permissions on the CreateRecyclePolicy, DescribeRecyclePolicies, DeleteRecyclePolicy, and ModifyRecyclePolicy APIs. You can contact the root account or CAM administrator to configure these permissions. Click Create Policy to configure a **recycling policy** and an **exclusion policy**. The **recycling policy** indicates that instances meeting recycling conditions will automatically stop releasing computing resources. The **exclusion policy** indicates that instances meeting exclusion conditions are not subject to the recycling policy. That is, such instances will not be forcibly stopped even if the conditions for idle recycling are met.

Configuring a Recycling Policy

The platform supports configuring conditions for instance resource metrics based on the CPU utilization, memory utilization, and GPU utilization. These metric conditions can be logically combined using either an **ANY** or **ALL** operator. When **ANY** is selected, the instance automatically stops when any configured metric condition is met. When **ALL** is selected, the instance automatically stops only when all configured metric conditions are met.

As shown in the figure below, when the following policy is configured:
The instance automatically stops when any of the following metric conditions are met, and this situation remains for more than 1 hour:
IF CPU utilization <= 10%
IF memory utilization <= 10%
IF GPU utilization <= 10%
This indicates that a dev machine will automatically stop when the CPU utilization, memory utilization, or GPU utilization of the instance remains less than or equal to 10% for more than 1 hour.

©2013-2026 Tencent Cloud. All rights reserved.

The platform supports the filtering of excluded instances by **instance name**. That is, the dev machines configured in this exclusion policy will not be subject to the aforementioned recycling policy. As shown in the figure above, the dev machine myNotebook will not be subject to the configured recycling policy. That is, the instance will not be forcibly stopped even if the recycling conditions are met.

Instance List

1. When an idle recycling policy is configured for this region, the following prompt message will be displayed above the instance list. By clicking **Automatic Stop Policy**, you can view the configured metric conditions and duration. In this case, you need to have permissions on DescribeRecyclePolicies.
2. When a dev machine automatically stops due to idleness, its status changes to Stopped. In addition, when you hover your mouse over the instance, the prompt message Automatic stop of idle instances triggered is displayed, as shown below.

Alarm Guide

Events related to the recycling of idle dev machines have been connected to Tencent Cloud EventBridge. The currently connected events are as follows:

- **devInstanceAutoReleasedOnIdle:** When a dev machine automatically stops due to idleness, this alarm event will be sent.
- **devInstancetobeReleasedOnIdle:** This alarm event will be sent 30 minutes and 24 hours before the dev machine is recycled, respectively.

For the detailed guide on configuring event alarms, see Operation Steps. This guide uses task events as an example. The configuration method for dev machine events is the same, and only requires replacement with the aforementioned dev machine events.

©2013-2026 Tencent Cloud. All rights reserved.

Using Lifecycle Scripts

Last updated: 2026-02-26 15:34:42

Rules for Lifecycle Script Configuration

The lifecycle configuration provides shell scripts that run when you create or start a dev machine. These scripts help you install custom dependencies and personalize the dev machine environment.
The lifecycle configuration adheres to the following rules:

- Creation script: a script that runs when a dev machine is started after it is created for the first time. The script runs only once.
- Startup script: a script that runs every time a dev machine is started, including when it is created for the first time.
- After being Base64-encoded, each script cannot exceed 16,384 characters.
- Each script will run as a root user.
- The $PATH environment variable for each script is `/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin`.
- Each script runs for a maximum of 5 minutes. The dev machine will fail to start when the time limit is exceeded. Avoid installing large dependency packages in scripts. The failure reason, such as startup script timeout, can be viewed on the instance details page.
- If an error occurs in a script, the dev machine will also fail to start. The failure reason can be viewed on the details page.
- If a script is copied from the editor to the TI-ONE web page, ensure that the editor used to edit the script uses the Unix-style orchestration.

Best Practices for Lifecycle Scripts

The following shows a practical case of using the lifecycle configuration:

- A lifecycle script runs as a root user, and a dev machine process runs as a TI-ONE user. To switch between users, you can run sudo -u tione in the script to switch to the TI-ONE user.
- Dev machines use conda to manage multiple kernels. You can activate conda env to install dependency packages for different kernels.

For example, to install the Python dependency package Fire in the kernel of conda_python3, you can write the startup script as follows:

```
#!/bin/bash
sudo -u tione -i <<'EOF'

# This will affect only the Jupyter kernel called "conda_python3".
```

©2013-2026 Tencent Cloud. All rights reserved.

```
  source /opt/conda/bin/activate python3

  # Replace fire with the name of the package you want to install.
  pip install fire
  # You can also perform "conda install" here as well.

  source /opt/conda/bin/deactivate

  EOF
```

For example, to install the Fire dependency package in all kernels, you can write the script as follows:

```
  #!/bin/bash
  sudo -u tione -i <<'EOF'

  # Note that "base" is special environment name, include it there as
well.

  for env in base /opt/conda/envs/*; do
      source /opt/conda/bin/activate $(basename "$env")

      # Installing packages in the Jupyter system environment can affect
stability of your tione
      # Notebook Instance.  You can remove this check if you'd like to
install Jupyter extensions, etc.
      if [ $env = 'JupyterSystemEnv' ]; then
        continue
      fi

      # Replace myPackage with the name of the package you want to
install.
      pip install fire
      # You can also perform "conda install" here as well.

      source /opt/conda/bin/deactivate
  done

  EOF
```

©2013-2026 Tencent Cloud. All rights reserved.

©2013-2026 Tencent Cloud. All rights reserved.

Using the Podman Image Tool in a Dev Machine

Last updated: 2026-02-26 15:35:31

Scenarios

This document demonstrates how to perform operations such as pulling, building, pushing, or running an image in a dev machine.

Using Podman in a Dev Machine

The Podman tool has been built in the dev machine to perform basic image operations. The commands for using Podman are basically consistent with those for using Docker.

Logging In to an Image Repository

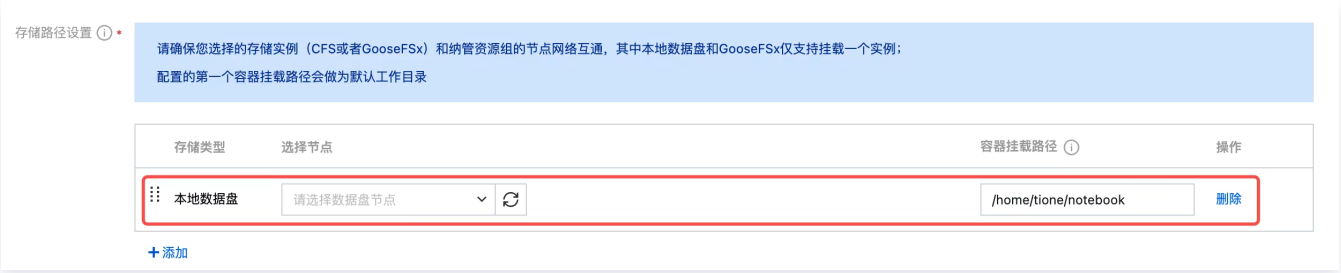
```
# Log in to a private image repository.
podman login <registry-url> -u <username> -p <password>
# Example:
podman login tione.tencentcloudcr.com --username ******* --password ******
```

Pulling an Image

```
# Pull an image from a repository.
podman pull <image-name>:<tag>
# Example:
podman pull tione.tencentcloudcr.com/tione-public-images/notebook-conda-
cpu:ubuntu22.04-miniforge3-py38
```

The process of pulling an image may require a large amount of disk space. If the node where the dev machine is located has a local data disk, it is recommended to mount the local data disk before pulling the image:

- Create or edit the dev machine and mount the local data disk to a container.



©2013-2026 Tencent Cloud. All rights reserved.

- Modify the graphroot storage path of the image layer or container read-write layer in the storage configuration file. For details, see Modifying Key Configurations of Podman.

Viewing an Image

```
# List local images.
podman images
# View the details of an image.
podman inspect <image-id|image-name>
# Example:
podman inspect tione.tencentcloudcr.com/tione-public-images/notebook-
conda-cpu:ubuntu22.04-miniforge3-py38
```

Building an Image

To reduce storage usage, it is recommended to build an image in stages.

```
# Build an image using a Dockerfile.
podman build -t <image-name>:<tag> <context-path>
# Example:
podman build -t myapp:1.0 .
```

Pushing an Image

```
# Tag an image.
podman tag <local-image> <registry>/<repository>:<tag>
# Push the image to a repository.
podman push <registry>/<repository>:<tag>
# Example:
podman tag myapp:1.0 registry.example.com/myteam/myapp:1.0
podman push registry.example.com/myteam/myapp:1.0
```

Running a Container

Currently, the feature of specifying a GPU to run an image is not supported.

```
# Run a container.
podman run [options] <image-name>
# Options: -d: run a container in the backend; -v: run a container on a
volume mount; --name: specify a container name.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
# Example:
podman run -d --name myContainer tione.tencentcloudcr.com/tione-public-
images/notebook-conda-cpu:ubuntu22.04-miniforge3-py38

# View the running containers. Options: -a: Include the stopped
containers.
podman ps
```

Deleting an Image or a Container

```
# Delete an image.
podman rmi -f <image ID or name> # -f means forced deletion.
podman image prune -a # -a: Delete unreferenced images.

# Delete a container.
podman rm -f <container ID or name> # -f means forced deletion.
podman container prune # Delete all exited containers.

# If the Podman storage is corrupted or needs to be completely cleaned up,
you can directly delete the graphroot and runroot directories.
# Example: rm -rf /var/lib/containers/storage/.
```

Modifying Key Configurations of Podman

Path of the storage configuration file: /etc/containers/storage.conf.

```
# Temporary storage path of a container runtime.
runroot = "/var/run/containers/storage"

# Storage path of the image layer or container read-write layer.
graphroot = "/var/lib/containers/storage"
```

For more Podman instructions, see the official documentation.

©2013-2026 Tencent Cloud. All rights reserved.

Model Management
Introduction to Model Management

Last updated: 2026-01-23 17:20:46

Model Management is a module provided by Tencent Cloud TI-ONE Platform (TI-ONE) for model management and optimization, comprising two sub-modules: Model Repository and Model Optimization.
Model Repository is the unified entry for model management on TI-ONE, allowing you to manage models trained on the platform as well as user-owned local models.

- You can import trained models and perform CRUD operations on third-party models.
- You can delete, modify, or view models that have been accelerated through the Model Optimization module.
- You can perform service release packaging operations on the managed models and their corresponding inference code and configuration files.
- You can manage models, model versions, and sub-directories under these versions.
- You can view online services associated with models.
- You can view batch prediction tasks associated with models.
- Features related to model hot updates are supported.

Model Optimization is a module of TI-ONE for inference acceleration. This module uses TI Acceleration Service (TI-ACC) capabilities to optimize the inference acceleration of the models in Model Repository, while reducing costs and improving efficiency. The models optimized through this module can only be used for inference services in Model Services of TI-ONE.

- You can perform inference acceleration on the models in Model Repository.
- You can quickly accelerate models in batches.
- You can accelerate common models in fields such as Computer Vision (CV), natural language processing (NLP), and Optical Character Recognition (OCR).
- Two optimization levels are supported: lossless and FP16.
- You can optimize models with multiple input nodes and fixed and dynamic shapes.
- The acceleration report for the model inference part can be output.
- You can save optimized models to Model Repository for service release.

©2013-2026 Tencent Cloud. All rights reserved.

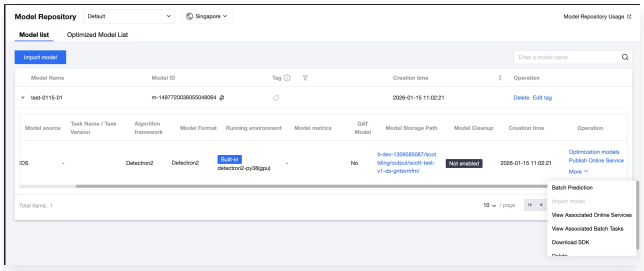
Usage of Model Repository

Last updated: 2024-12-12 22:06:42

Model Repository supports importing models. The list page manages different models and their versions. The import window allows you to import models directly through task-based modeling and upload models from Cloud Object Storage (COS).

Model Repository Lists

The Model Repository lists display models and their corresponding versions, allowing you to perform operations such as releasing online services. The Model Repository lists are divided into the model list and the optimized model list. The model list displays the original models you uploaded, while the optimized model list displays the optimized models saved after model optimization.



- You can click **Delete** to delete a model and its corresponding version.
- You can click **Edit Tag** to edit a model tag.
- You can click **Optimization Models** for a model version to redirect to the task creation page of Model Optimization to optimize the model version.
- You can click **Publish Online Services** for a model version to redirect to the service creation page of Model Services to release and package the current model service.
- You can click **Batch Predictions** for a model version to redirect to the task creation page of Batch Predictions for batch predictions.
- You can click **Import Model** for a model version to add directories and sub-versions under the directory of the current model version. Currently, this feature only supports importing sub-versions for the Savedmodel format, mainly used for hot updates in TensorFlow Serving.
- You can click **View Associated Online Services** to view the model services associated with a model version and their statuses.
- You can click **View Associated Batch Tasks** to view the batch tasks associated with a model version and their statuses.
- You can click **Download SDK** to open COSBrowser for downloading. You need to install the COSBrowser client first.

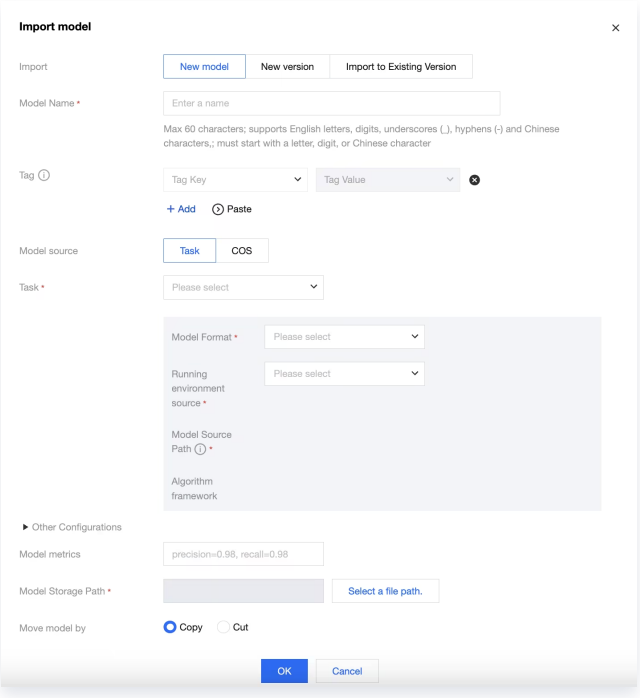
©2013-2026 Tencent Cloud. All rights reserved.

- You can click **Delete** to delete a model version. Note that the deleted versions cannot be recovered, model versions that have been associated with service groups and whose services are running cannot be deleted, and model versions that have been associated with batch prediction tasks and whose tasks are running cannot be deleted. Before deletion, you can click **View Associated Service Groups** or **View Associated Batch Tasks** to check whether the deletion operation can be performed.

Importing a Model

Choose **Model Management** > **Model Repository**, and click **Import Model** to start importing a model.

Importing a New Model



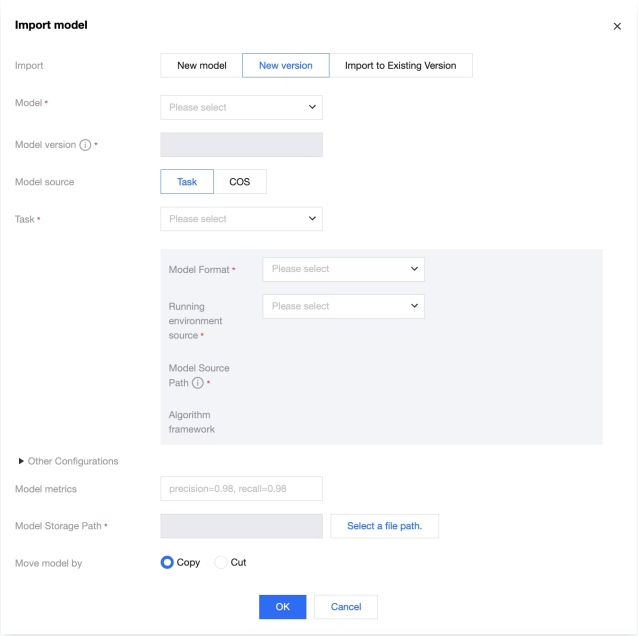
1. Enter the model name and configure a custom tag as required. (Tags are used to categorize resources from different dimensions. You need to configure Tag Settings first in the console. You can also click **Usage Guide for Tag Management** to view tag settings.)
2. Import a model from a task:
   - Select the name of the training task.
   - After selecting the training task, select the corresponding model format. If the task has the corresponding algorithm framework, model source path, and model metric information, they will be automatically populated. Note that the populated model metric information is the latest reported metric information.
   - Select the built-in running environment or pull a running environment from Tencent Container Registry (TCR) through custom selection. Currently, optimized models can only be run in the built-in running environment.

©2013-2026 Tencent Cloud. All rights reserved.

- The model file of this task will be automatically obtained. You can process the model file by cutting or copying. After processing, the COS path will be automatically generated for you.

3. Import a model from COS:
   - Select a model format. Currently, supported model formats include Savedmodel, Frozen Graph, TorchScript, Detectron2, PyTroch native, PMML, ONNX, MMDetection, and Hugging Face.
   - Enter the desired model metrics corresponding to the model you want to upload.
   - Select a built-in running environment or pull a running environment from TCR through custom selection. If you select a built-in running environment, you need to select the corresponding model format mentioned above.
   - Upload the model file. For details, see Model Package Specifications.

Importing a New Version



1. Select a model and import its new version. The model version number will automatically increase by 1 based on the existing highest version number, for example, V2 or V3.
2. Import a model from a task:
   - Select the name of the training task.
   - After selecting the training task, select the corresponding model format. If the task has the corresponding algorithm framework, model source path, and model metric information, they will be automatically populated. Note that the populated model metric information is the latest reported metric information.
   - Select the built-in running environment or pull a running environment from Tencent Container Registry (TCR) through custom selection. Currently, optimized models can only be run in the built-in running environment.

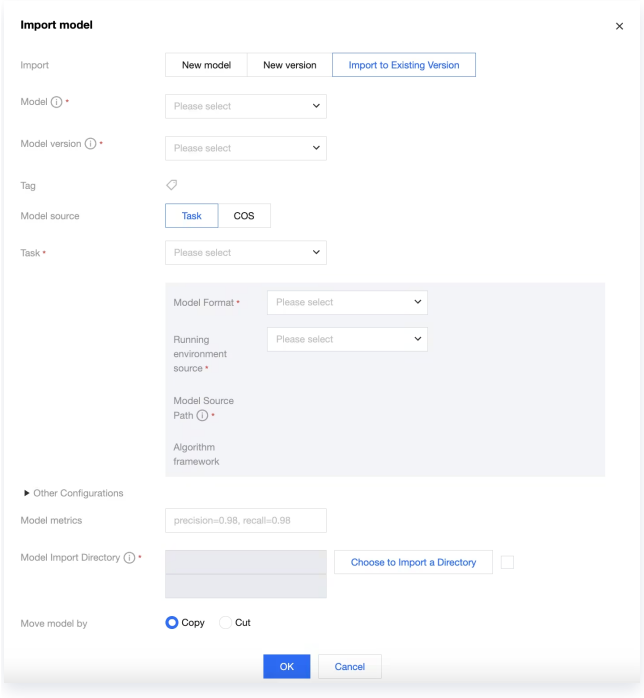
©2013-2026 Tencent Cloud. All rights reserved.

- The model file of this task will be automatically obtained. You can process the model file by cutting or copying. After processing, the COS path will be automatically generated for you.

3. Import a model from COS:
   - Select a model format. Currently, supported model formats include Savedmodel, Frozen Graph, TorchScript, Detectron2, PyTroch native, PMML, ONNX, MMDetection, and Hugging Face.
   - Enter the desired model metrics corresponding to the model you want to upload.
   - Select a built-in running environment or pull a running environment from TCR through custom selection. If you select a built-in running environment, you need to select the corresponding model format mentioned above.
   - Upload model files: You need to upload the model file, inference script code, and service API configuration file. The inference code and configuration file should be named as config.json and model_service.py respectively. For details, see Model Package Specifications.

After completing the above configurations, click **OK**. The model is successfully imported.

Importing a Model to the Current Version



1. Select a model version and import its sub-version. Currently, only the Savedmodel format is supported, mainly used for hot updates in TensorFlow Serving. You can select models in the Savedmodel format and their versions existing in the model list.
2. Import a model from a task:
   - Select the name of the training task for the model in the Savedmodel format.

©2013-2026 Tencent Cloud. All rights reserved.

- After the training task is selected, if the task has the corresponding algorithm framework, model source path, and model metric information, they will be automatically populated. Note that the populated model metric information is the latest reported metric information.
- Select the built-in running environment or pull a running environment from Tencent Container Registry (TCR) through custom selection. Currently, optimized models can only be run in the built-in running environment.
- The model import directory is the storage directory appended under this model version. This directory must be the main directory of the model version or its sub-directory, and you can also choose to append a directory for the sub-version to the model version directory. The platform will automatically generate a new directory based on the timestamp, and you can choose whether to select this directory as needed.
- After model cleanup is enabled, the platform will search for the model directory under the model storage path corresponding to the model version, and retain the latest N models in the model directory based on the configured upper limit for the number of retained models.
- The model file of this task will be automatically obtained. You can process the model file by cutting or copying. After processing, the COS path will be automatically generated for you.

3. Import a model from COS:
   - Select a model format. Currently, only the Savedmodel format is supported.
   - Select a built-in running environment or pull a running environment from TCR through custom selection. If you select a built-in running environment, you need to select the corresponding model format mentioned above.
   - Select a model source path. You need to select the folder corresponding to the sub-version of the model you want to import.
   - Enter the desired model metrics corresponding to the model you want to upload.
   - The model import directory is the storage directory appended under this model version. This directory must be the main directory of the model version or its sub-directory, and you can also choose to append a directory for the sub-version to the model version directory. The platform will automatically generate a new directory based on the timestamp, and you can choose whether to select this directory as needed. All files in the selected model source path will be copied or pasted to the model import directory.
   - After model cleanup is enabled, the platform will search for the model directory under the model storage path corresponding to the model version, and retain the latest N models in the model directory based on the configured upper limit for the number of retained models.
   - All files in the selected model source path will be copied or pasted to the model import directory based on the model movement method you selected.

©2013-2026 Tencent Cloud. All rights reserved.

After completing the above configurations, click **OK**. The model is successfully imported into the existing version.

Model Optimization

In the Model Optimization module, you can perform inference acceleration on the models in the original model list. To perform inference acceleration on models, see Model Optimization.

Publishing an Online Service

In the model version operations corresponding to the original model list, you can click **Publish Online Service** for an original model to perform service release packaging operations. In the model version operations corresponding to the original model list, you can also click **Publish Online Service** for an optimized model to perform service release packaging operations.

©2013-2026 Tencent Cloud. All rights reserved.

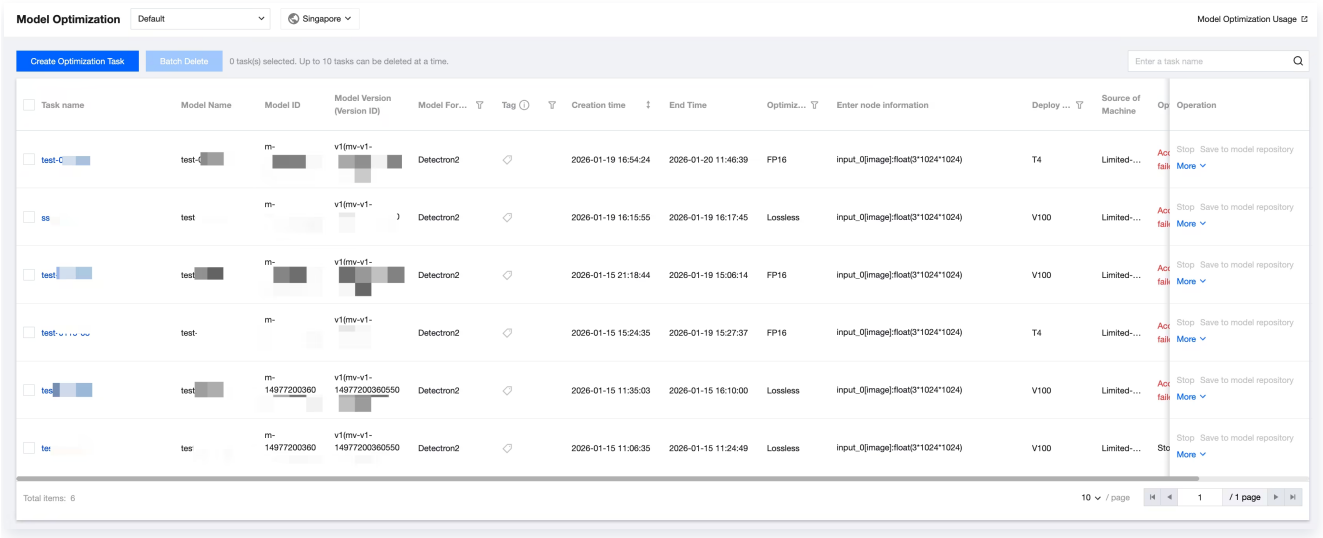
Usage of Model Optimization

Last updated: 2025-08-14 11:35:31

Model Optimization is a module of Tencent Cloud TI-ONE Platform (TI-ONE) for inference acceleration. This module uses TI Acceleration Service (TI-ACC) capabilities to optimize the inference acceleration of the models in Model Repository, while reducing costs and improving efficiency. It is currently free for a limited time, but the models optimized through this module can only be used for inference services in Model Services of TI-ONE.

Optimization Task List

The optimization task list allows you to view information such as the optimization task name, optimization progress, and acceleration ratio, and perform operations such as saving optimized models to Model Repository. The optimization progress provides statuses like Starting, Accelerating, and Acceleration Completed, and the optimization process usually takes several minutes to complete.

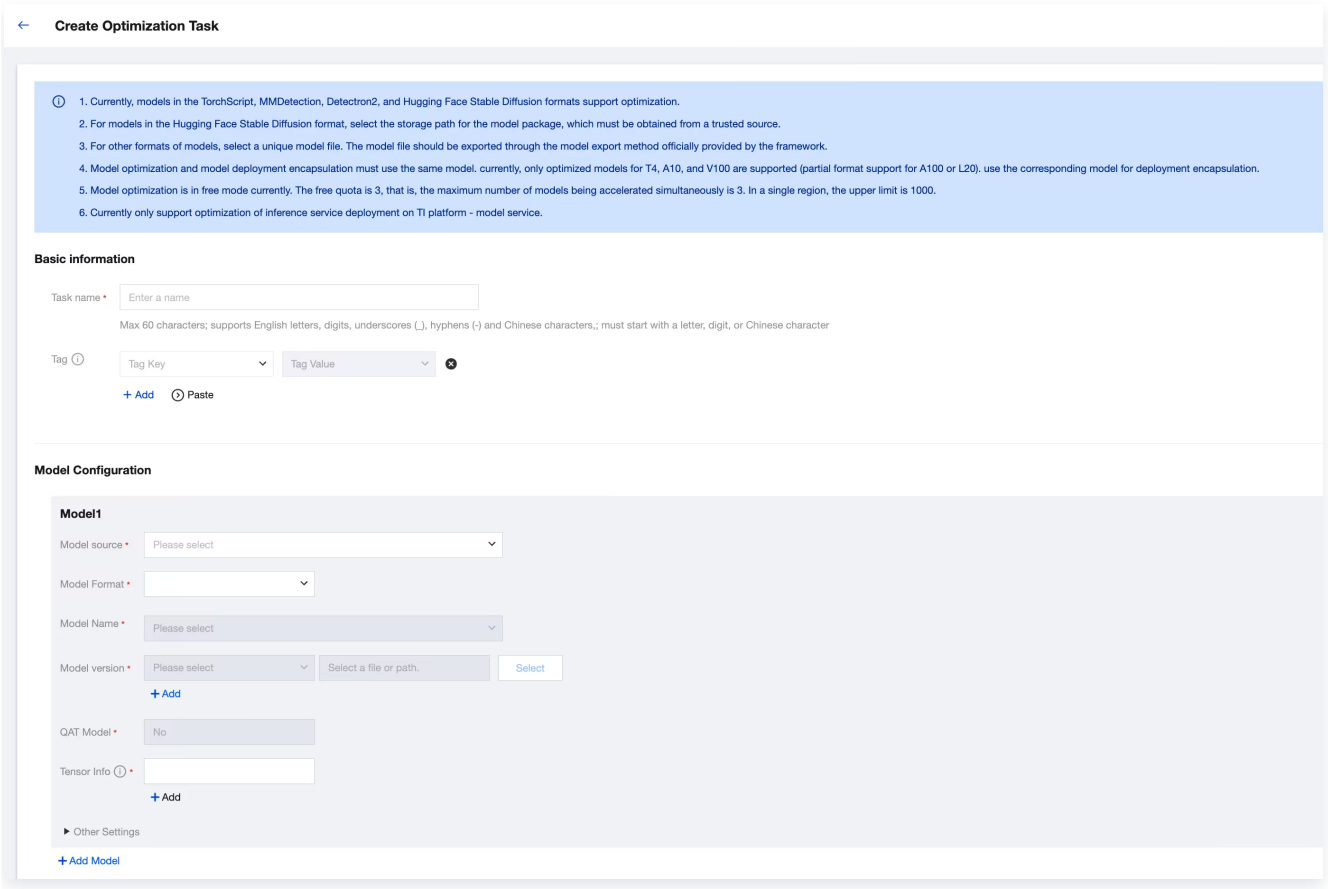


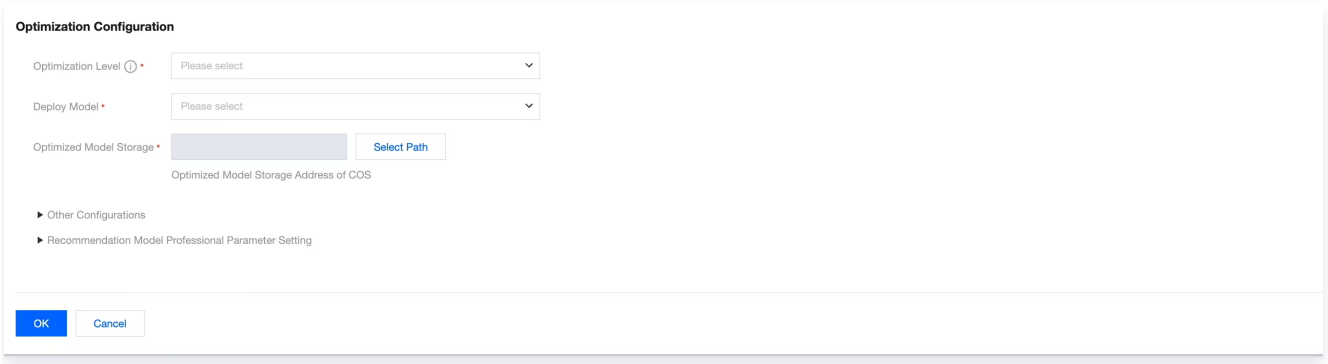
- You can click **Stop** to stop an optimization task in progress (Starting or Accelerating) immediately.
- You can save the task in the Acceleration Completed status for service release. You can click **Save to Model Repository** to immediately save the model optimized through this optimization task to the model optimization list page of Model Repository.
- You can re-accelerate the task that fails during acceleration. You can choose **More** > **Re-accelerate** to edit and accelerate the optimization task again.
- You can click **Delete** to delete an optimization task immediately.
- You can click **Edit Tag** to edit a tag.

Creating an Optimization Task

©2013-2026 Tencent Cloud. All rights reserved.

Currently, models in TorchScript, MMDetection, Detectron2, Hugging Face-Stable Diffusion, ONNX, Savedmodel, and Frozen Graph formats are supported. You can refer to the following guidance for integration.





1. Enter a **task name**. When only a model and its version are optimized, the task name should be the one you entered. When multiple models or versions are added, multiple optimization tasks will be generated, and the name of each task consists of the task name you entered + -model name + -model version + # + a sequence number.
2. Add a **tag** as needed.
3. Select a **model source**. The selected model source should be consistent with the model source field in Model Repository. The model name will be filtered based on the selected model source.

©2013-2026 Tencent Cloud. All rights reserved.

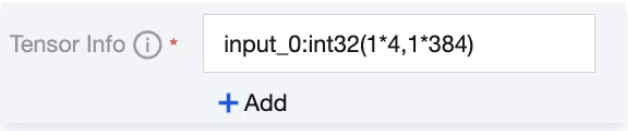
4. Select a **model name**, which refers to the name of the model to be optimized. You can add multiple models and versions for an optimization task. When multiple models or versions are added, multiple optimization tasks will be generated, and the name of each task consists of the task name you entered + -model name + -model version + # + a sequence number.
5. Select a **model version**, which refers to the version of the model to be optimized. You can select multiple versions and their corresponding files.
6. Select a **model file**, which refers to the specific file of the model to be optimized. Model files in different formats have different suffixes. For details, see the specific prompts.
7. **Quantization-Aware Training (QAT)** is a technique used in deep learning to train models that can be quantized and allow these models to be deployed on the hardware with limited computation capabilities. QAT simulates quantization during training and allows modules to adapt to lower bit-widths while maintaining accuracy. Unlike Post-Training Quantization (PTQ) which quantizes pre-trained models, QAT involves quantizing models during training.

   > **Note:**
   > The QAT process can be broken down into the following steps:
   > 1. Define a model: Define a floating-point model, which is identical to a general model.
   > 2. Define a quantized model: Define a quantized model that has the same structure as the original model but includes quantization operations (such as torch.quantization.QuantStubQ) and dequantization operations (such as torch.quantization.DeQuantstub()).
   > 3. Prepare the data: Prepare the training data and quantize it to the appropriate bit-width.
   > 4. Train a model: During training, use the quantized model for forward and backward passes, and calculate the accuracy loss using dequantization operations when each epoch or batch ends.
   > 5. Perform quantization again: During training, quantize the model parameters again using dequantization operations and continue training using new quantization parameters.
   > 6. Fine-tuning: Use the fine-tuning technology to further improve the model accuracy after the training is completed.

8. Fill in the **Tensor Information** field:
   - **Input data** should be considered as a whole. The shape information of **input data** must be provided in advance before model optimization, to ensure that the model optimization task is performed normally.
   - **Input data** uses multidimensional tensors as its basic elements, consisting of lists (list), dictionaries (dict), tuples (tuple), and their nested combinations.

©2013-2026 Tencent Cloud. All rights reserved.

- For each element in the input data, you need to provide its shape size. The shape size of each element can be represented in the **element:dtype(shape)** format, and each element occupies one line.
- **element** is represented as follows:
  - For Layer 1 of the input data:
    - If this layer is a list or tuple, each element is represented as input_0, input_1, ...; particularly, if the input data is a single tensor, it is represented as input_0.
    - If this layer is a dict, each element is represented as the key of the dict.
  - For Layer n (n > 1) of the input data:
    - If this layer is a list, each element is represented as [0], [1], ...
    - If this layer is a tuple, each element is represented as (0), (1), ...
    - If this layer is a dict, each element is represented as [key].
- **dtype** is the basic data type of an element, for example, float, int32, and array.char.
- **shape** is the actual shape of an element. There are 3 cases:
  - Fixed size: represented by a **size** value. **size** consists of n positive integers separated by asterisks (*). For example, **3*1024*1024** indicates that the element is a 3-dimensional tensor with a size of 3 x 1024 x 1024.
  - Dynamic and continuous size: represented by 2 **size** values separated by a comma (,). For example, **3*1024*2048,3*2048*2048** indicates that the element is a variable-sized 3-dimensional tensor, with a minimum size of 3 x 1024 x 2048 and a maximum size of 3 x 2048 x 2048.
  - Dynamic and discrete size: represented by a list of **size** values. For example, **[3*1024*1024,3*2048*2048]** indicates that the element is a variable-sized 3-dimensional tensor, with 2 possible sizes: either 3 x 1024 x 1024 or 3 x 2048 x 2048.
- **Example configurations:**
  - **Example 1**: The input data is a **single tensor**.
    - The tensor has a fixed size, its data type is float, and its actual shape size is 1 x 3 x 1024 x 1024.
    - The **Tensor Information** field is filled in with 1 line, namely, input_0:float(1*3*1024*1024).



  - **Example 2**: The input data is a **single-layer list**.
    - There are 3 elements, all of which are tensors with the data type int32, and their actual shape sizes are dynamic and continuous sizes with a minimum value of 1 x 4 and a maximum value of 1 x 384.

©2013-2026 Tencent Cloud. All rights reserved.

- The **Tensor Information** is filled in with 3 lines, namely, with input_0:int32(1*4,1*384), input_1:int32(1*4,1*384), and input_2:int32(1*4,1*384).



- **Example 3**: The input data is a **single-layer dict**.
  - There are 3 elements, their keys are input_ids, input_mask, and segment_ids respectively, their values are all tensors with the data type int32, and their actual shape sizes are dynamic and continuous sizes with a minimum value of 1 x 128 and a maximum value of 8 x 128.
  - The **Tensor Information** is filled in with 3 lines, namely, input_ids:int32(1*128,8*128), input_mask:int32(1*128,8*128), and segment_ids:int32(1*128,8*128).



9. The latest **acceleration library engine version** is used by default, and the latest acceleration library engine version varies for each format.
10. **Optimization Level** can be **FP16** or **Lossless**. Lossless indicates that the optimization is performed based on the original model accuracy, while FP16 indicates that the optimization is performed based on the FP16 accuracy. It is recommended to use FP16 to perform the optimization because it offers faster inference than the lossless method and generally does not cause accuracy degradation.
11. The deployment instance type can be **T4**, **V100, or A10**, and **A100 or PNV5b** is supported only for certain model formats. When **PNV5b** is selected, **Resource Group** is required. When T4 is used, you can select **Time-Limited Free (Pay-as-You-Go)** or **Resource Group**. If you select other instance types, only the free mode is supported. The deployment instance type selected during optimization must be consistent with the instance type of the actual deployment service. That is, if a T4 deployment service is needed, the instance type should be set to T4 here.
12. You can click **Select Path** to select the location for storing the optimized model. This path cannot overlap with the main directory of the original model. When you click **Save to Model**

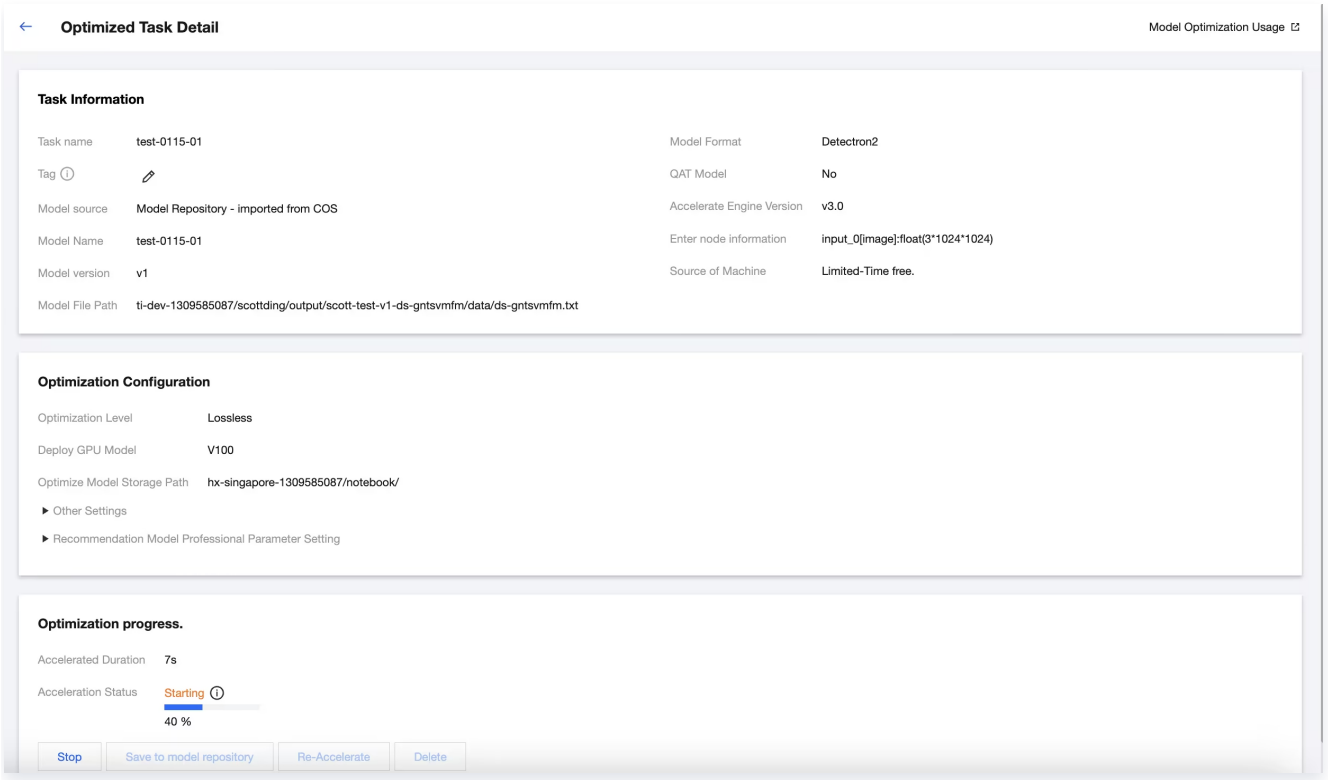
©2013-2026 Tencent Cloud. All rights reserved.

Repository for a model in the Acceleration Completed status, all files in the original model directory will be copied to a newly generated m-xxx/mv-xxx folder under this path, and the optimized model tiacc.pt will be generated and stored in the model folder. If you want to deploy the optimized model, Model Package Specifications of TI-ONE must be followed.

13. After you click **OK**, the optimization task is performed, and the optimization task list page is displayed.

Optimized Task Details

You can click the **optimized task name** to go to the optimization task details page. On this page, you can view the detailed task information and optimization report, and perform corresponding operations.



©2013-2026 Tencent Cloud. All rights reserved.

Model Package Specifications

Last updated: 2026-01-23 17:21:56

Structure of the Model Package Directory

Structure of the original model package directory:

```
cos://bucket/train_models/mv-2302330437837096
├── model
|   ├── model.pth
└── requirements.txt
```

Structure of the optimized model package directory (When an optimized model is saved to Model Repository, an m-xxx/mv-xxx folder will be generated under the storage path for the optimized model. Under this folder, the original model package is copied to generate a new optimized model file model/tiacc.pt.):

```
cos://bucket/optimized_models_save/m-230233043/mv-2302330437837096
├── model
|   ├── model.pth
|   ├── tiacc.pt (an optimized model automatically generated by the Model
Optimization module of Tencent Cloud TI-ONE Platform, TI-ONE in short)
└── requirements.txt
```

- model: directory for storing models. You can specify this directory but need to ensure that model_service.py can be loaded correctly.
- model.pth: original model file. You can specify the file name but need to ensure that model_service.py can be loaded correctly.
- tiacc.pt: optimized model file. This file can be automatically generated after the Model Optimization feature is used. If you need to use an optimized model for inference and service deployment, you need to add the logic for this model to read the load in the model_service.py inference script.
- requirements.txt: third-party dependency package. If the inference script has third-party dependencies, they can be placed in the requirements.txt file. If not, you can ignore this parameter. Before Tencent Kubernetes Engine (TKE) is started, pip will be used to install third-party dependencies in the requirements.txt file.
- Others: other relevant dependency files. They can be randomly placed and packaged for uploading.

©2013-2026 Tencent Cloud. All rights reserved.

©2013-2026 Tencent Cloud. All rights reserved.

Model Evaluation
Introduction to Model Evaluation

Last updated: 2026-01-15 13:29:53

Model Evaluation provides a wizard-based evaluation task submission method for LLM evaluation. Tencent Cloud TI-ONE Platform (TI-ONE) supports two evaluation methods, namely manual evaluation and automated evaluation.

- **Manual evaluation:** provides the manual scoring feature after the inference of the model to be evaluated and supports manual evaluation for the model effect.
- **Automated evaluation:** No manual participation is required during the entire process. TI-ONE performs automated evaluation based on built-in open-source evaluation sets and automatic metrics (such as pass@1, ROUGE, and F1) or user-defined uploaded open-source evaluation sets and user-defined metrics. Automated evaluation also allows you to conduct a **quick test** on the effect of the trained model during training, as well as service deployment for checkpoint models. You can also interact with an LLM in a dialog box to evaluate its effect.

> **Note:**
> Manual evaluation can be usually used together with automated evaluation when best practices are implemented. For example, during the model development process, you can perform automated evaluation first based on open-source data sets (which can be extended to the proprietary standardized data sets accumulated within the current enterprise), to obtain a relatively good result. Then, you can use manual evaluation to verify the model effect again in the final stage before model release, or evaluate its effect at any time after the model has been launched.

For models after manual evaluation and automated evaluation, **a visual comparison of model effects** is supported. Multiple models and metrics can be horizontally compared through a radar chart to provide an intuitive display of effect comparison.

©2013-2026 Tencent Cloud. All rights reserved.

Quick Test

Last updated: 2026-01-23 17:43:02

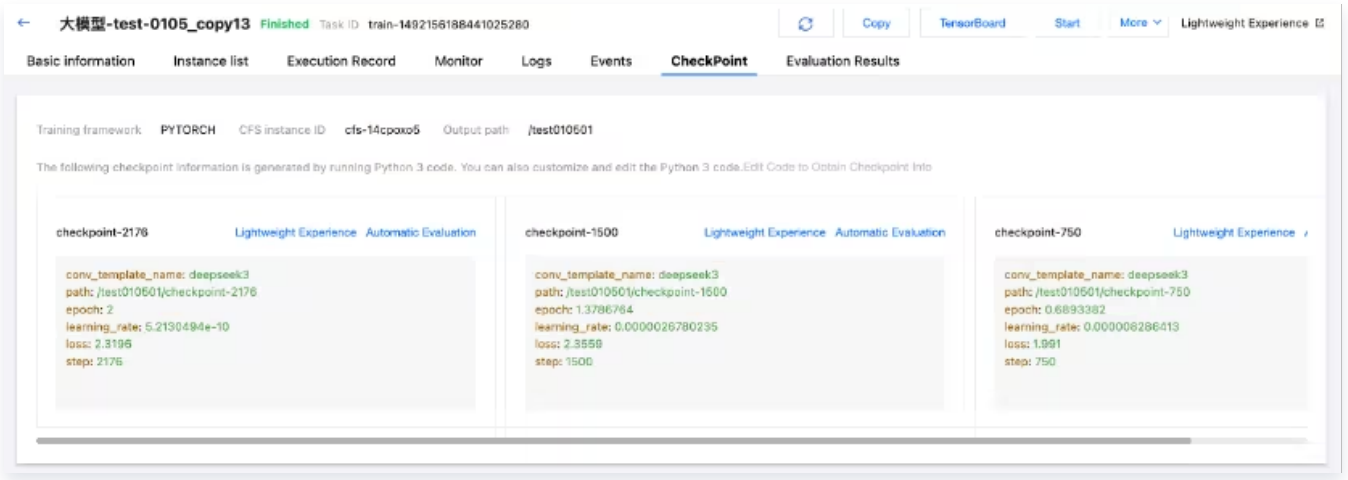
During the training process, you can conduct a quick test on the trained model to initially check its effect.

Prerequisites

You have created the task through Task-based Modeling and completed the checkpoint output.

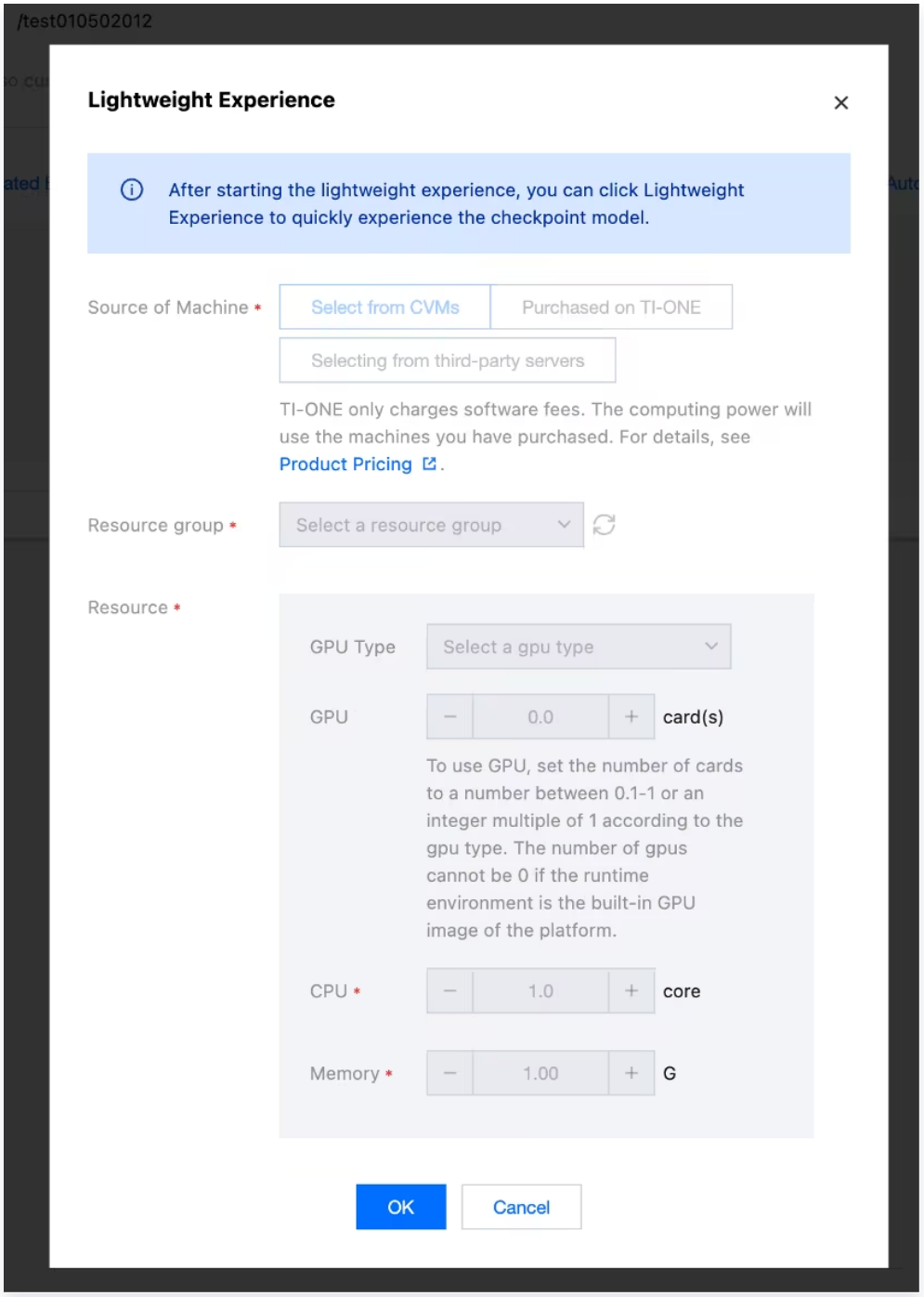
Operation Steps

1. Log in to the Platform and choose **Training Workshop > Task-based Modeling** in the left sidebar to go to the task list page.
2. On the task page, click the **task name** to go to the task details page.
3. Click the **CheckPoint** tab and select the checkpoint card on which you want to conduct a quick test to preview the model effect.



4. Click **Quick Test**. In the pop-up window, select the required resources from the resource group used during training.

©2013-2026 Tencent Cloud. All rights reserved.



5. After the quick test goes through the creating, starting, and ready states, you can click **Quick Test**. In the pop-up window, check the model effect.
6. In the pop-up window, you can send questions and chat with the model to experience the model effect.

©2013-2026 Tencent Cloud. All rights reserved.

Automated Evaluation

Last updated: 2026-03-16 16:23:53

Overview

Automated evaluation provides a wizard-based approach for submitting evaluation tasks to build evaluation tasks. It allows you to quickly conduct general effectiveness tests based on the built-in evaluation sets of the platform or start evaluation tasks by using custom evaluation sets and evaluation metrics. Its detailed features are as follows:

- **Entry 1:** creating an automated evaluation task under Task-based Modeling > CheckPoint
  - You can conduct a lightweight evaluation on the checkpoint of a training task. To enable automated evaluation, you only need to configure the basic task information and select a built-in evaluation set provided by the platform.
  - You can view evaluation results.
- **Entry 2:** creating an automated evaluation task on the Automated Evaluation tab
  - You can quickly create an evaluation task through built-in open-source evaluation sets and automatic metrics (such as pass@1, ROUGE, and F1).
  - You can initiate an evaluation task by using custom evaluation sets and evaluation metrics. Three evaluation modes are supported: Evaluation Only, Inference and Evaluation, and Custom Evaluation.

Evaluation Mode	Mode Description	Metric Configuration and Result Output
Evaluation Only	You can upload an evaluation set with model inference results and complete the scoring feature in the Automated Evaluation module.	In the two modes, you can customize evaluation metrics, debug metrics, and view the overall results and a specific evaluation result: • Customize metrics: When you customize evaluation metrics, you need to configure the scoring method of each metric. For example, when you use a judge model for scoring, you need to configure the judge model and scoring prompts, and customize preprocessing and postprocessing scripts to process inputs and outputs to obtain metric results. • Debug metrics: Before you formally initiate an evaluation task, you can conduct a small number of evaluations on the prediction sample. During debugging, you can adjust scoring prompts
Inference and Evaluation	When you upload an evaluation set containing only queries (questions), you can output the inference results and complete	

©2013-2026 Tencent Cloud. All rights reserved.

	the scoring in the Automated Evaluation module.	and preprocessing and postprocessing scripts to achieve the expected evaluation results. • View the overall results: You can view the evaluation results of each model on each evaluation set. • View a specific evaluation result: You can view the result of each scoring step
Custom Evaluation (This feature is in beta testing.)	You can perform evaluations through custom evaluation images. Evaluation sets, custom images, storage mounts, and other elements can be combined into a task configuration. Each configuration includes evaluation sets, image selection, version selection, mount path settings, startup commands, parameter settings, and environment variables. By selecting images and versions only or selecting images and versions and then configuring mount paths, you can implement evaluations using images only or using images and mount paths.	You can customize evaluation metrics and output evaluation results using images or by mounting evaluation scripts. For specific usage, see 2.2 Configuring a Task.

Prerequisites

When you create an automated evaluation task, you need to prepare an evaluation set (use a built-in set or upload a custom set) and the model or service to be evaluated.

Preparing an Evaluation Set

- Using a built-in evaluation set

For specific built-in evaluation sets, see Introduction to Built-in Open-Source Evaluation Sets.

©2013-2026 Tencent Cloud. All rights reserved.

- Uploading a custom evaluation set

  Unlike the method for directly using a built-in open-source evaluation set of the platform, after you prepare an evaluation set, you need to enter the path of the evaluation set in Cloud File Storage (CFS)/GooseFSx/data sources when you create a task.

  Download and decompress the objective data set of OpenCompass.
  Copy the decompressed objective data set of OpenCompass to the local objective data set folder.
  During evaluation, enter the data set on your CFS instance, for example, `/test_data/xsum`.

Models or Services to Be Evaluated

- Models to be evaluated
  - Built-in LLMs

    Certain built-in LLMs in Model Hub supported by the Automated Evaluation module
  - Self-owned LLMs

    After you prepare the model to be evaluated, you need to enter the path of the evaluation set in CFS/GooseFSx/data sources when you create a task. For details on how to obtain the path, see the instructions in **Uploading a Custom Evaluation Set under Preparing an Evaluation Set** mentioned above.
- Models or services to be evaluated
  - Select models from Online Services.

    During evaluation, you can select models to be evaluated from Online Services of Tencent Cloud TI-ONE Platform (TI-ONE). You need to start the model to be evaluated as a service on TI-ONE in advance. For details on the deployment guide, see Online Services Deployment.
  - Enter a service address.

    You can enter a service address during evaluation. You need to prepare the service in advance and record its address.

Creating an Evaluation Task

There are 2 entries for creating an evaluation task. You can create an automated evaluation task under Task-based Modeling > CheckPoint or on the Model Evaluation tab.

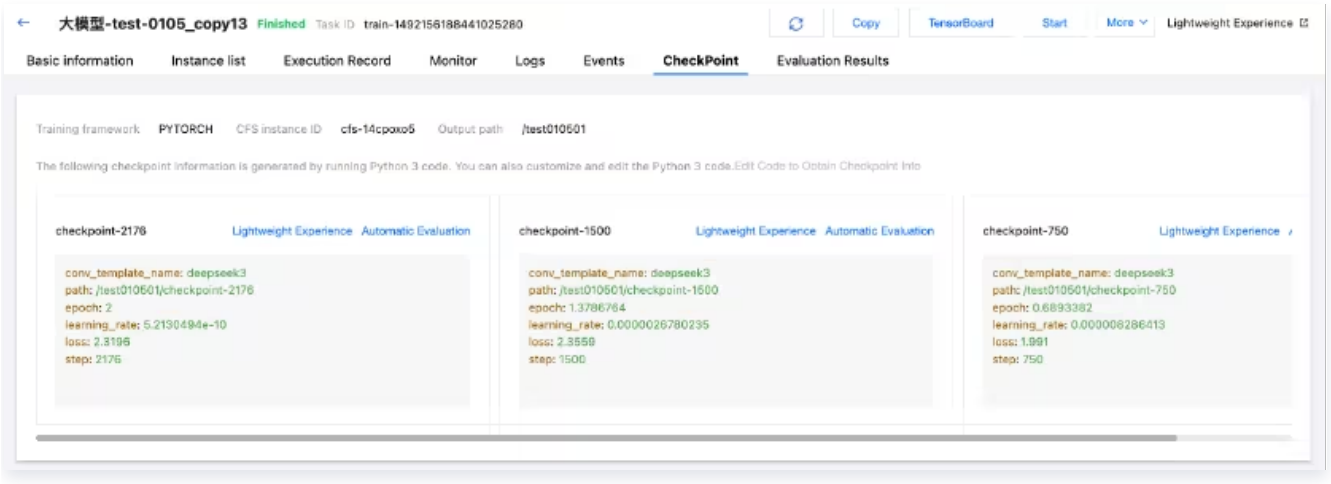
Entry 1: Creating an Automated Evaluation Task Under Task-based Modeling > CheckPoint

©2013-2026 Tencent Cloud. All rights reserved.

Overview: Log in to the TI-ONE console, and choose **Training Workshop > Task-based Modeling** in the left sidebar to go to the task list page. On the task page, click the **task name** to go to the task details page. The prerequisite for using this entry is that you have created a task through Task-based Modeling and completed the checkpoint output.

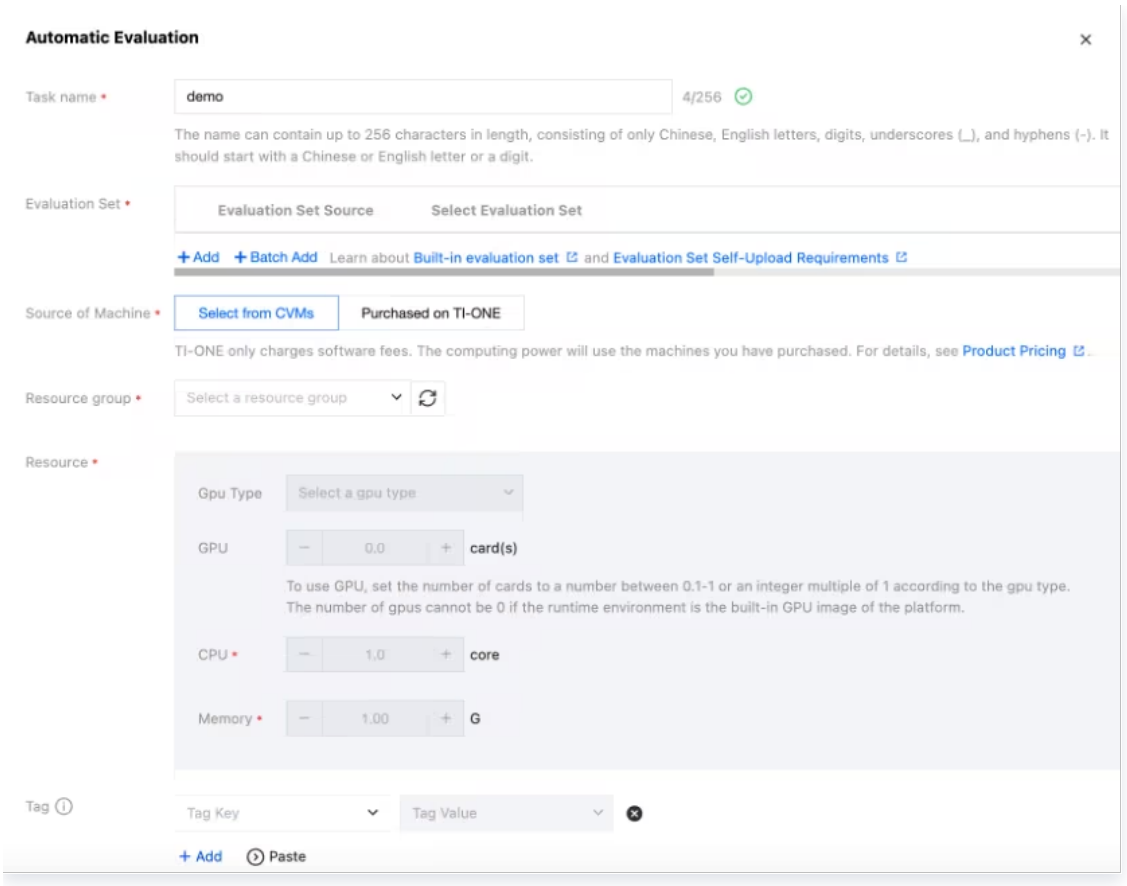
1. Configuring the Basic Information and an Evaluation Set

Click the **CheckPoint** tab and select the checkpoint card on which you want to conduct a quick test to preview the model effect.



Click **Automated Evaluation**. In the pop-up window, enter the task name, and select a built-in evaluation set of the platform and the required resources. Both pay-as-you-go and yearly/monthly subscription billing modes are supported. Currently, you can only perform evaluations on built-in evaluation sets.

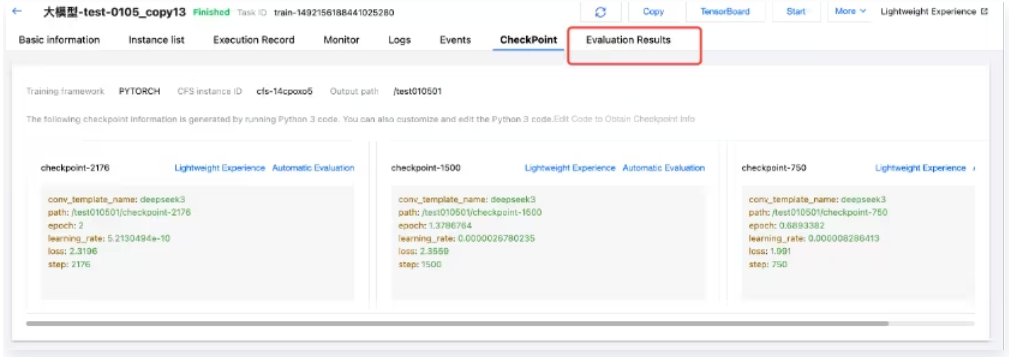
©2013-2026 Tencent Cloud. All rights reserved.



2. Viewing Evaluation Results

After entering the information, click **Create**. The task will enter the Automated Evaluation – Inference in Progress status. Please wait patiently.

Click **View Automated Evaluation Results** after the automated evaluation is completed.

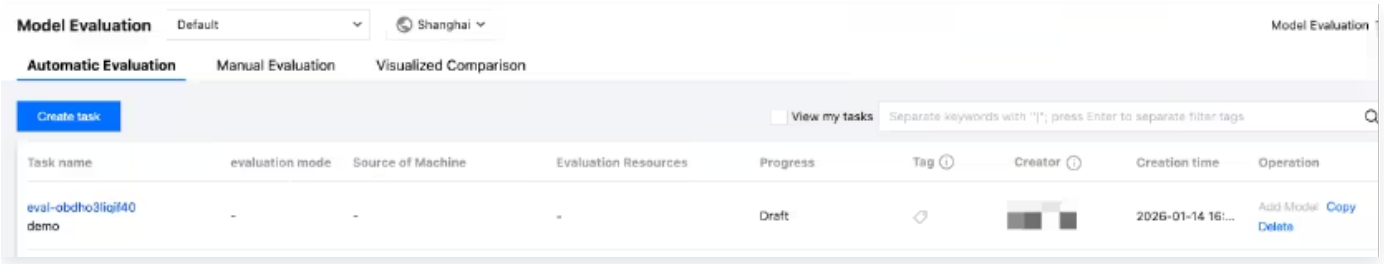


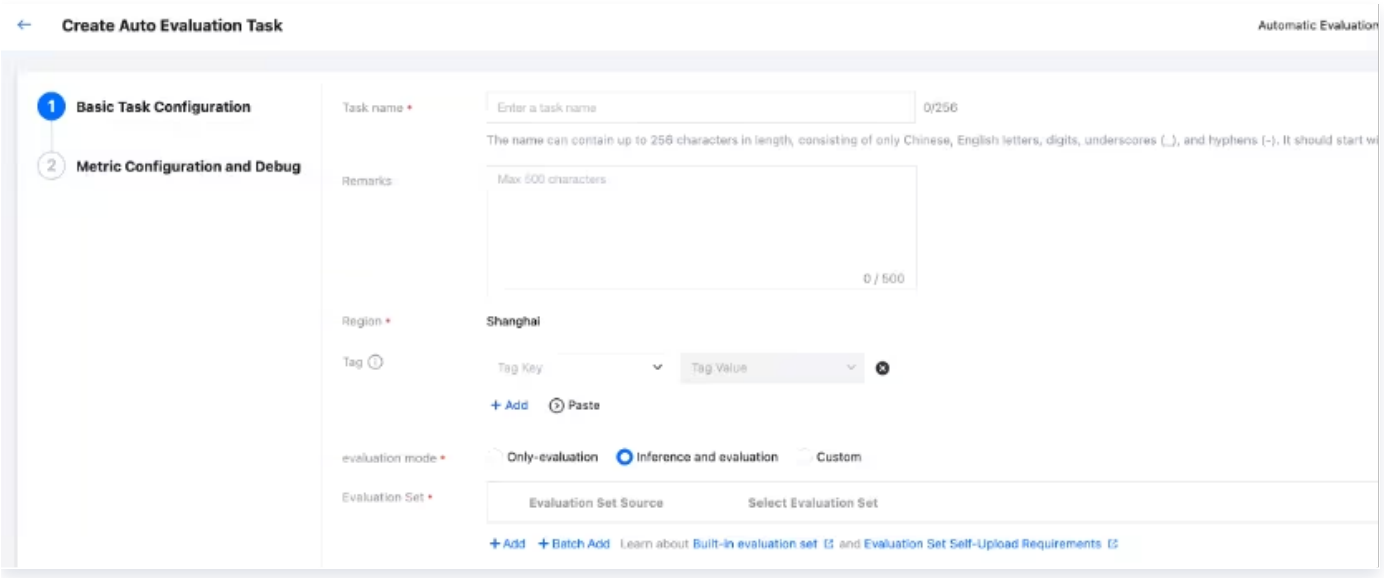
Entry 2: Creating an Automated Evaluation Task on the Automated Evaluation Tab

Overview: Log in to the TI-ONE console, choose **Model Services > Model Evaluation** in the left sidebar, and then click the **Automated Evaluation** tab to go to the task list page. Then click **Create Task** to go to the task creation page.

©2013–2026 Tencent Cloud. All rights reserved.

- To create an automated evaluation task in the Evaluation Only mode or Inference and Evaluation mode, perform two steps: Step 1 is to configure a basic task. In this step, you need to select the evaluation set, the model to be evaluated, and the evaluation resources. Step 2 is to configure and debug metrics. In this step, you need to customize evaluation metrics and their configurations, perform debugging before task submission, and initially view results.
- To create an automated evaluation task in the Custom Evaluation mode, you need to complete the basic task configurations and select the model to be evaluated along with the evaluation resources.





1. Evaluation Only and Inference and Evaluation Modes

1.1 Configuring a Basic Task

Parameter	Description
Task Name	Name of an automated evaluation task. You can enter it based on the rules according to the interface prompts.
Remarks	Add remarks to a task as needed.
Region	Services under the same account are isolated by region. The value of the Region field is automatically entered based on the region you selected on the

©2013-2026 Tencent Cloud. All rights reserved.

	service list page.
Tag	Used for permission isolation between evaluation tasks.
Billing	• Billing mode: You can select the pay-as-you-go mode or yearly/monthly subscription (resource group) mode only when the evaluation mode is Inference and Evaluation: ○ In the pay-as-you-go mode, you do not need to purchase a resource group in advance. Fees are charged based on the CVM instance specifications on which the service depends. When the service is started, fees for the first two hours are frozen. After that, fees are charged hourly based on the number of running instances. ○ In the yearly/monthly subscription (resource group) mode, you can use the resource group deployment service purchased from the Resource Group Management module. Since computing resource fees are already paid when the resource group is purchased, no fees need to be charged when the service is started. • Resource group: If you select the yearly/monthly subscription (resource group) mode, you can select a resource group from the Resource Group Management module.

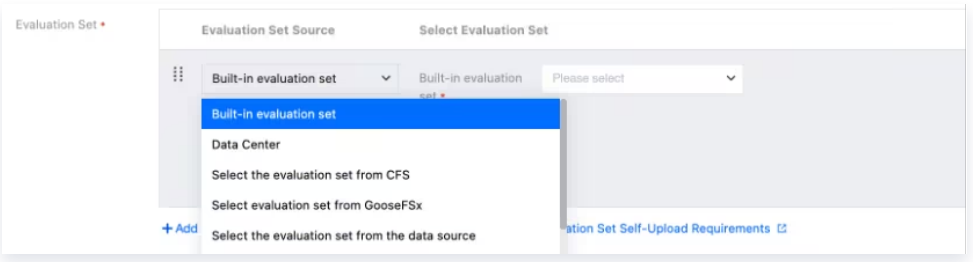
1.2 Configuring an Evaluation Set

When the evaluation mode is Evaluation Only or Inference and Evaluation, you need to configure an evaluation set to perform evaluations. The evaluation set should be selected from built-in evaluation sets/Data Center/CFS/GooseFSx/data sources/resource groups. You can configure the parameters of an evaluation set and preview the configured evaluation set.

- Select an evaluation set source.
  - If you select a built-in evaluation set, you can directly enable the evaluation with one click.
  - If you select an evaluation set from Data Center, you need to choose **Platform Data Center > Data Set** to mount the corresponding data set first. For evaluation sets in Data Center, you can filter them by Business Tag to improve the efficiency for selecting an evaluation set.
  - If you select an evaluation set from data sources, you need to choose **Platform Management > Data Source Management** to create a data source. Note: The data source mount permissions are divided into read-only mount and read-write mount. For a data source for which you need to output training results, you can configure its mount permission to read-write.
  - If you select an evaluation set from CFS or GooseFSx, you need to select the CFS instance or GooseFSx instance from the drop-down list and enter the data source directory to be mounted by the platform. The last layer of the path is the folder directory, for example, `/test_data/ceval`. When the directory is mounted, you need to ensure that

©2013-2026 Tencent Cloud. All rights reserved.

the data set in this directory is the one to be evaluated. Otherwise, the evaluation will fail. Currently, you cannot specify a specific file name when you enter the path.



- Configure the parameters of an evaluation set and preview the configured evaluation set.
  - Configure parameters: You can configure inference hyperparameters and judge model scoring parameters in batches for an evaluation set.

The inference hyperparameters are described as follows:

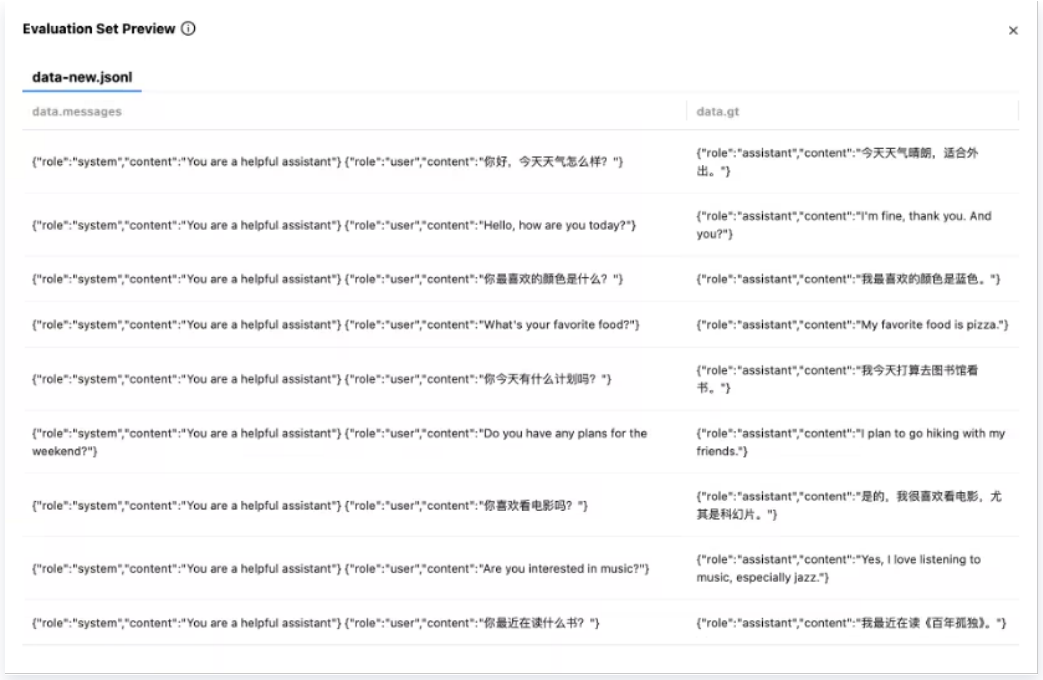
repetition_penalty: controls the repetition penalty.
max_tokens: controls the maximum length of the output text.
temperature: A higher temperature makes outputs more random; a lower temperature makes outputs more focused and deterministic.
top_p and top_k: control the diversity of the output text. Higher values produce more diverse outputs. It is recommended to configure only 1 of the temperature, top_p, and top_k parameters.
do_sample: specifies the sampling method for model inference. When this parameter is set to true, the sample method is used; when this parameter is set to false, the greedy search method is used, and the top_p, top_k, temperature, and repetition_penalty parameters do not take effect.

The judge model scoring parameters are described as follows:

MAX_JUDGING_CONCURRENCY: indicates the maximum number of requests that can be sent to a judge model simultaneously during scoring after each model to be evaluated has completed inference on the current data set. Setting this value too low may cause the throughput of a model to decrease and lead to a long evaluation time, while setting this value too high may cause request timeout issues.
MAX_JUDGING_RETRY_PER_Q: indicates the maximum number of retries for each piece of data when an exception occurs during scoring, such as a network failure or a scoring request queuing timeout. If the value is 0, no retry is performed. Note that too many retries may result in a longer evaluation time.
INFERENCE_COUNT: number of inference data entries generated by default.

  - Data Preview: You can preview the selected data set.

©2013-2026 Tencent Cloud. All rights reserved.



1.3 Configuring a Model or Service to Be Evaluated

You need to select the model or service to be evaluated to generate inference results for evaluation. Sources for models to be evaluated include built-in LLMs/training tasks/CFS/GooseFSx/data sources/resource groups, while sources for model services include online services/fill-in service addresses. When you configure models/services to be evaluated, the parameter setting feature is provided.

- Select a model:
  - If you select a built-in LLM, you can quickly enable an evaluation.
  - If you select a model from training tasks, you can select a training task in the current region or the checkpoint of the task.
  - If you select a model from data sources, you need to choose **Platform Management > Data Source Management** to create a data source. Note: The data source mount permissions are divided into read-only mount and read-write mount. For a data source for which you need to output results, you can set its mount permission to read-write.
  - If you select a model from CFS or GooseFSx, you need to select the CFS instance or GooseFSx instance from the drop-down list and enter the directory to be mounted by the platform.
- Select a service:
  - If you select a service from Online Services, you need to select the online service name. To ensure that the evaluation can be initiated successfully, make sure that the service is running normally. In addition, you need to select the corresponding authentication information.
  - If you choose to enter a service address, you need to enter the complete service call URL along with the corresponding authentication information. If you use this method to select

©2013-2026 Tencent Cloud. All rights reserved.

a service, you can check whether the service can run normally through the connectivity test.

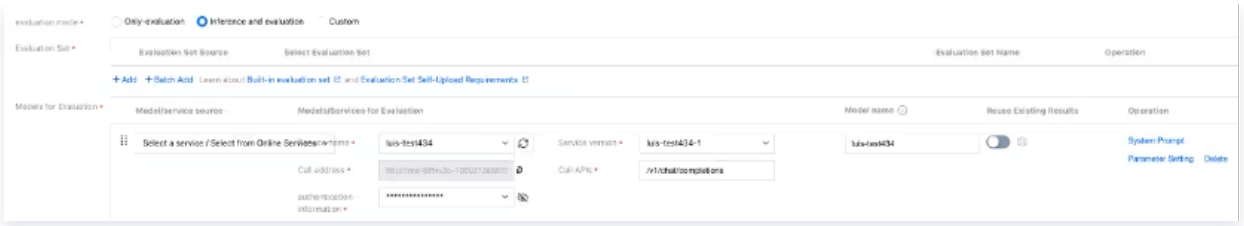
- Configure parameters: You can configure inference hyperparameters, startup parameters, and performance parameters.
  - Configure inference hyperparameters as follows:

    repetition_penalty: controls the repetition penalty.
    max_tokens: controls the maximum length of the output text.
    temperature: A higher temperature makes outputs more random; a lower temperature makes outputs more focused and deterministic.
    top_p and top_k: control the diversity of the output text. Higher values produce more diverse outputs. It is recommended to configure only 1 of the temperature, top_p, and top_k parameters.
    do_sample: specifies the sampling method for model inference. When this parameter is set to true, the sample method is used; when this parameter is set to false, the greedy search method is used, and the top_p, top_k, temperature, and repetition_penalty parameters do not take effect.
  - Configure startup parameters: More details in Service Deployment Parameter Configuration Guide. MAX_MODEL_LEN is the default parameter configured on the platform, which specifies the maximum number of tokens a model can process in a single inference operation. Its default value is 8192 on the platform. If you set this parameter to a very high value upon startup, GPU out-of-memory or performance degradation issues may occur. You can adjust this value appropriately based on task requirements.
  - Configure performance parameters: MAX_CONCURRENCY and MAX_RETRY_PER_QUERY are the default parameters configured on the platform.

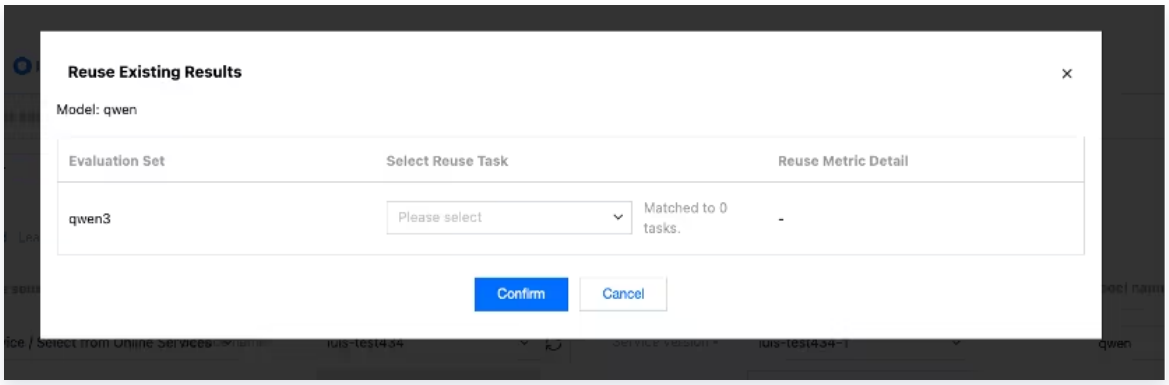
    MAX_CONCURRENCY indicates the maximum number of requests that can be sent to a model simultaneously during evaluation. Setting this value too low may cause the throughput of a model to decrease and lead to a long evaluation time, while setting this value too high may cause GPU out-of-memory or request timeout issues. Its default value is 24 on the platform. You can adjust this value appropriately based on task requirements.
    MAX_RETRY_PER_QUERY indicates the maximum number of retries for each piece of data when an exception occurs in requesting the inference service, such as a request timeout or a network failure. If the value is 0, no retry is performed (default value: 0). You can adjust this value appropriately based on task requirements.
- You can reuse previous evaluation results.

  When you select a model to be evaluated, you can select the evaluation results of a previous evaluation task with the same model name and the same evaluation set name.

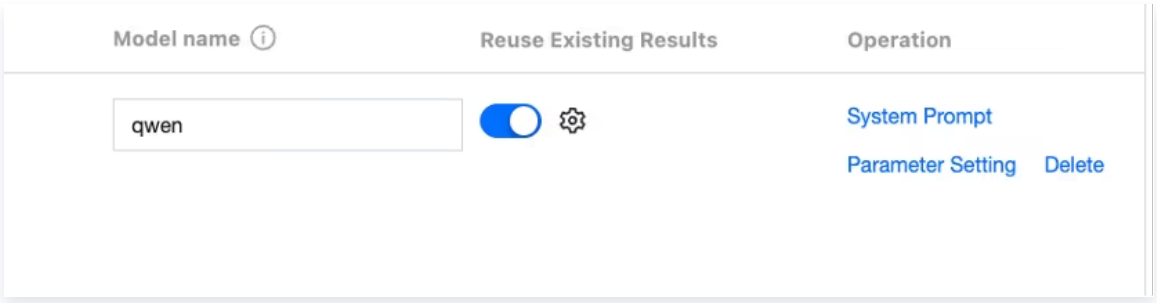
©2013-2026 Tencent Cloud. All rights reserved.



Enable the reuse switch and select the evaluation tasks to be reused.



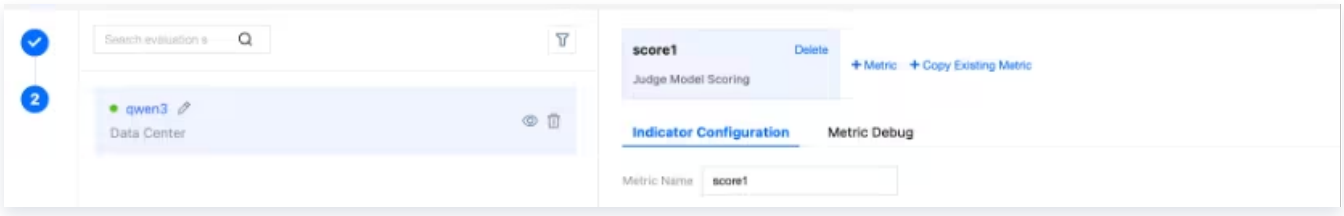
Directly reuse results without the need to perform inference and scoring again.



1.4 Configuring Metrics

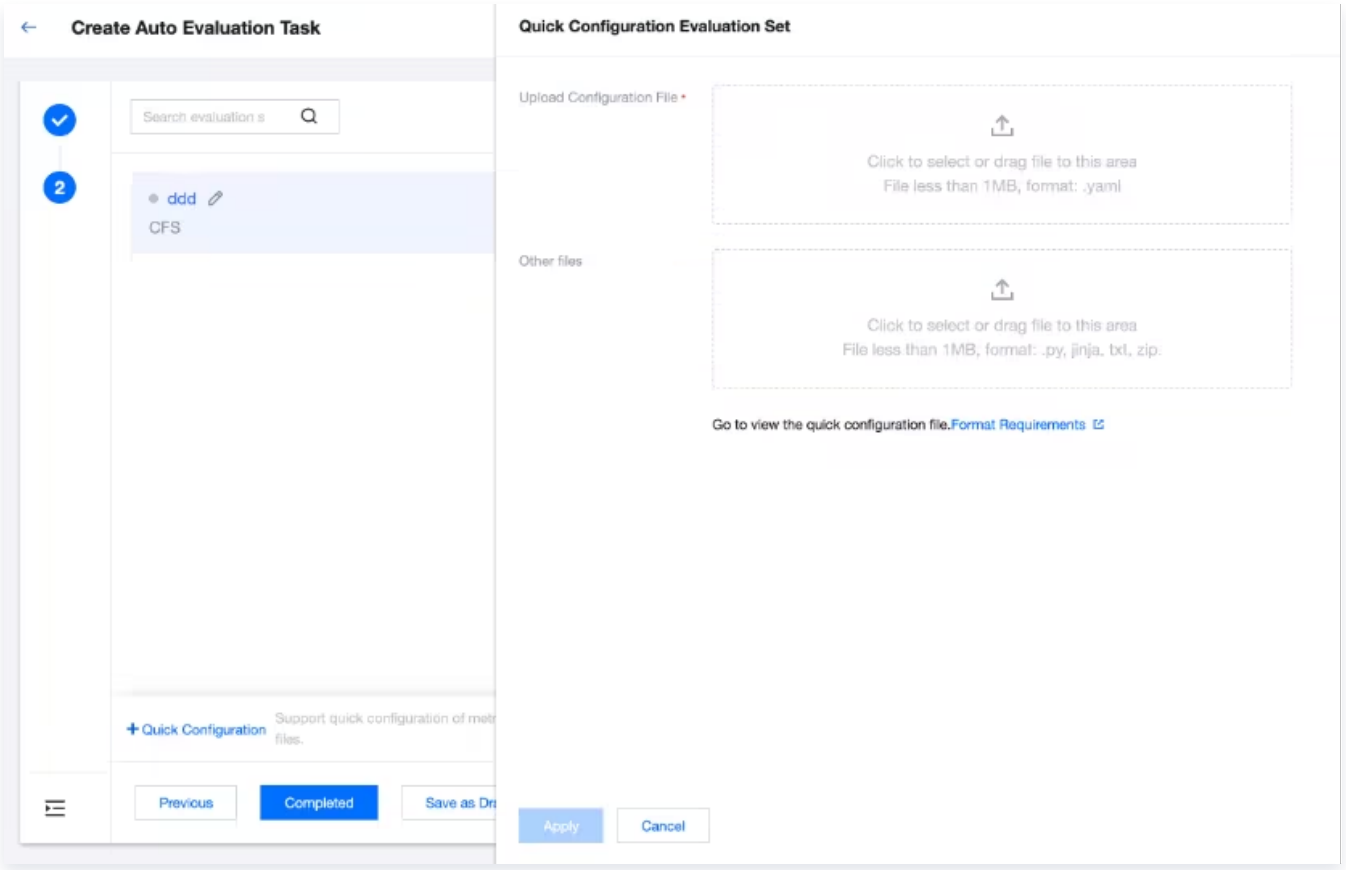
For built-in evaluation sets, you do not need to configure metrics because the platform provides the default metrics for automated evaluation. For custom evaluation sets, you need to configure metrics. You can use the judge model scoring method to customize these metrics. In addition, you can configure preprocessing and postprocessing scripts to highly customize the inputs and outputs of the data to achieve the desired evaluation results.

- Add metrics by clicking **+Metric** or **+Copy Existing Metric**. After metrics are added successfully, you need to configure them. During configuration, you can customize an evaluation process. The evaluation process includes preprocessing, judge model scoring, and postprocessing. For usage instructions, see Prompt for Judge Model Scoring and Preprocessing/Postprocessing Format Requirements .



©2013-2026 Tencent Cloud. All rights reserved.

- Configure evaluation metrics quickly through **Quick Configuration**. If you use the quick configuration feature, you need to upload files containing an evaluation set, metric names corresponding to the evaluation set, and detailed configuration information (such as the judge model information, scoring prompts, and preprocessing and postprocessing scripts) corresponding to each metric name. You can click Quick Configuration and upload your custom YAML configuration file and the files that need to be referenced. After the upload is completed, you can click **Apply**. The platform will automatically populate the information on the page based on the metrics in the YAML configuration file corresponding to the evaluation set name you entered, as well as the configuration information of these metrics. For details on how to upload files when you use the quick configuration feature, see Quick Configuration File Specifications for Metrics.



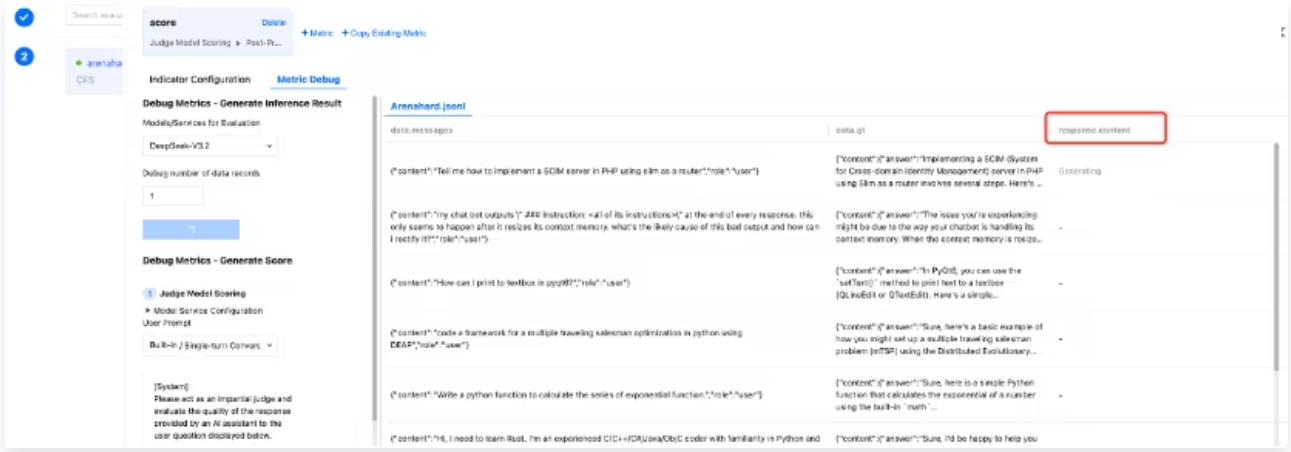
1.5 Debugging Metrics

For metrics configured for custom evaluation sets, you can debug them before a formal evaluation is initiated. You can view automatically generated responses (inference results) and scoring results to determine whether the scoring prompts and processing scripts need to be adjusted. During debugging, you can select a configured model service or model to be evaluated.

- Generate the answer from the model to be evaluated.
  - Select a service to be evaluated.

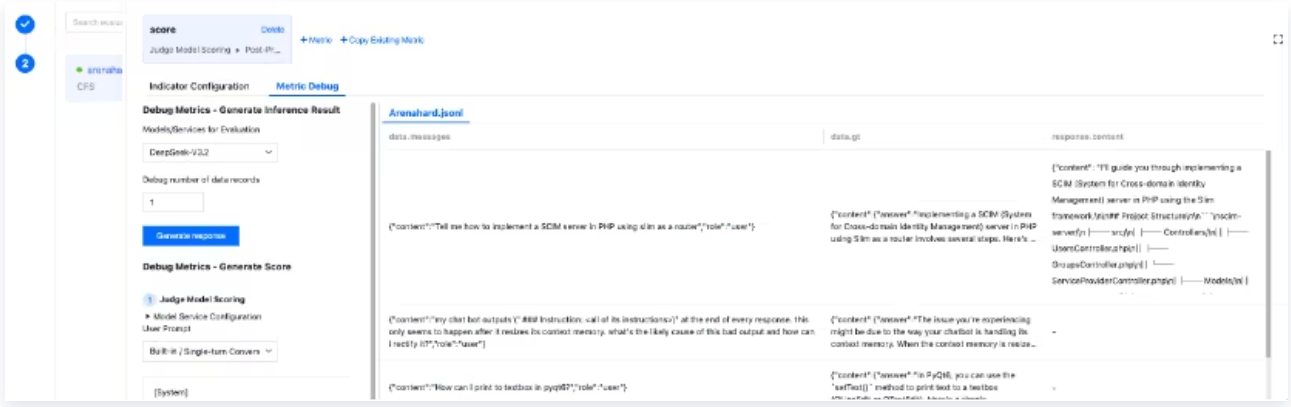
    Click **Generate Response** to automatically generate the response column.

©2013-2026 Tencent Cloud. All rights reserved.



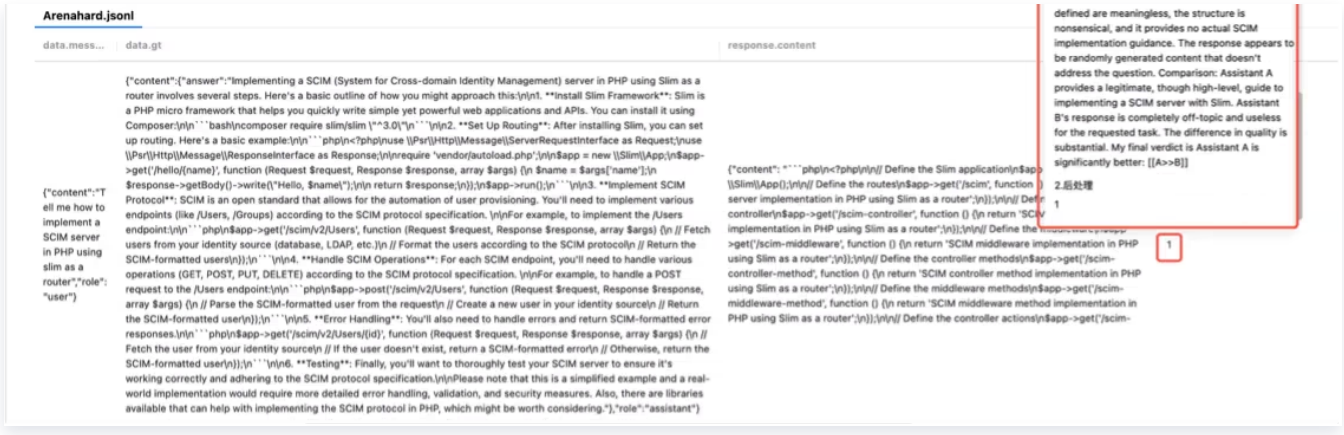
- Select a model to be evaluated.

  Manually enter response. The manually entered response will be automatically scored.



- Generate scoring results.

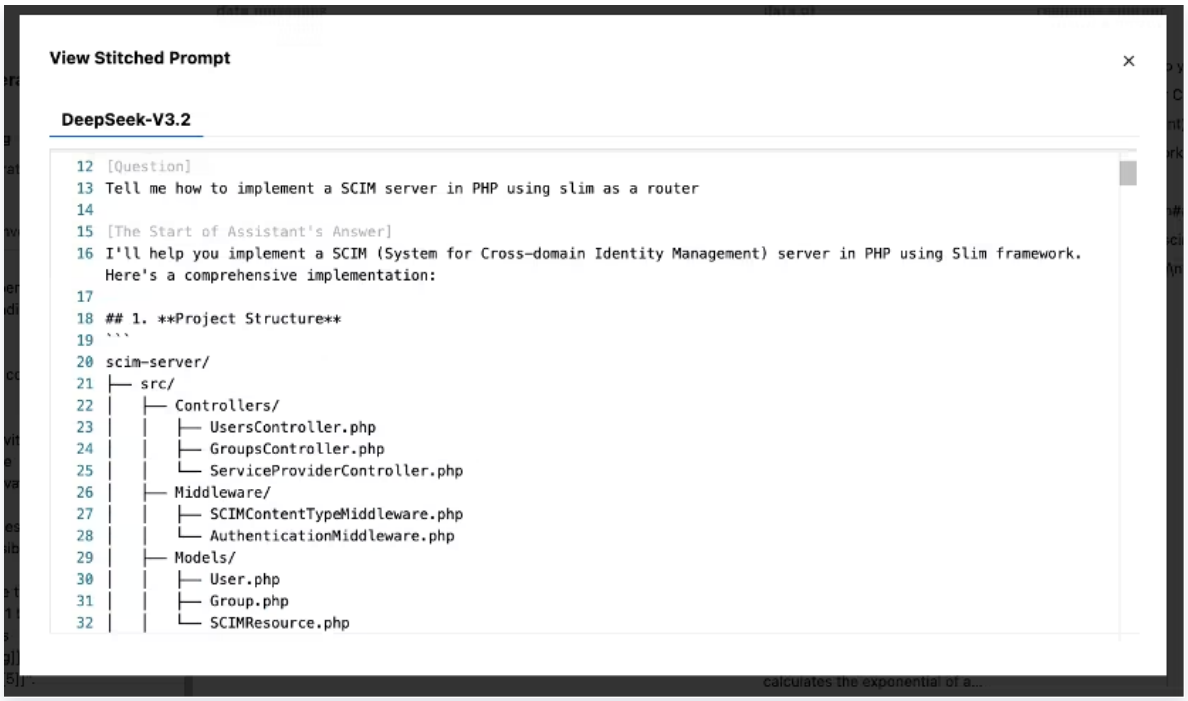
  Click **Generate Scoring Results** to automatically generate the scoring result column. You can hover your mouse pointer to view the results of each scoring step.



- View a concatenated prompt.

  Click **View Concatenated Prompt** to view the effect of the concatenated prompt.

©2013–2026 Tencent Cloud. All rights reserved.

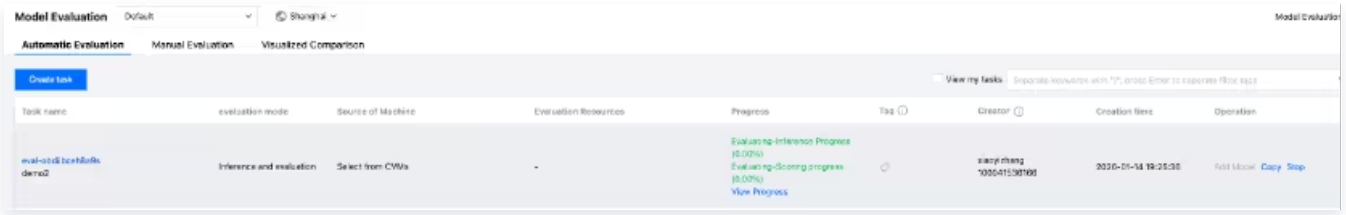


1.6 Viewing Evaluation Results

After you enter the above information, click Finish to start the evaluation task. After an evaluation task is successfully created, the following information will be displayed on the task list page: Task Name, CVM Instance Source, Evaluation Resources, Progress, Tag, Creator, Creation Time, and Operations (Stop, Restart, Delete, and Copy).

- Overall evaluation progress

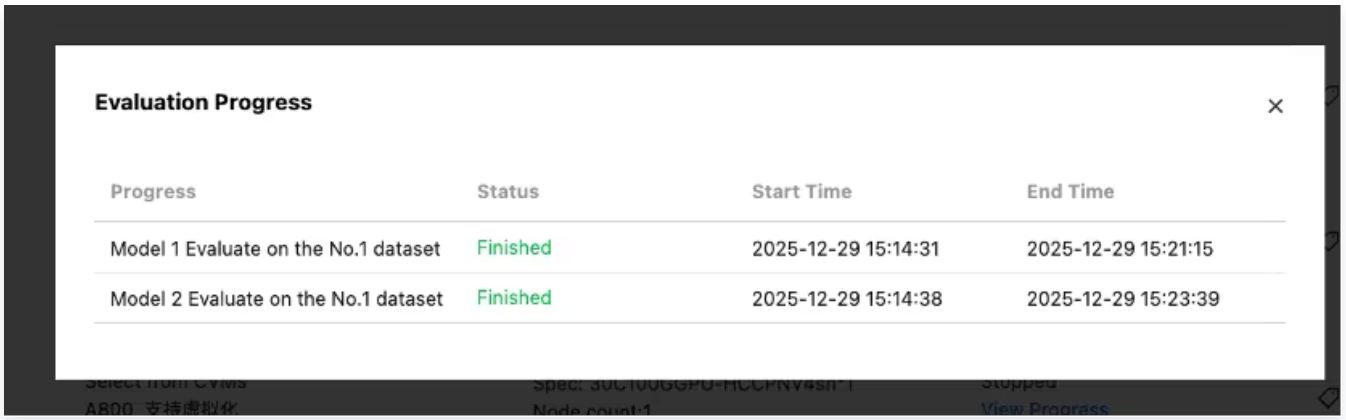
  During the evaluation process, you can view the progress:



> **Note:**
> - Formulas for calculating the progress: Total number of evaluation (inference/scoring) data entries = Number of models x Number of evaluation data entries. Progress N% = Number of evaluated (inference/scoring) data entries/Total number of evaluation (inference/scoring) data entries%.
> - During the evaluation process, if any data fails to be evaluated, the remaining data will continue to be evaluated until all the data has been evaluated.

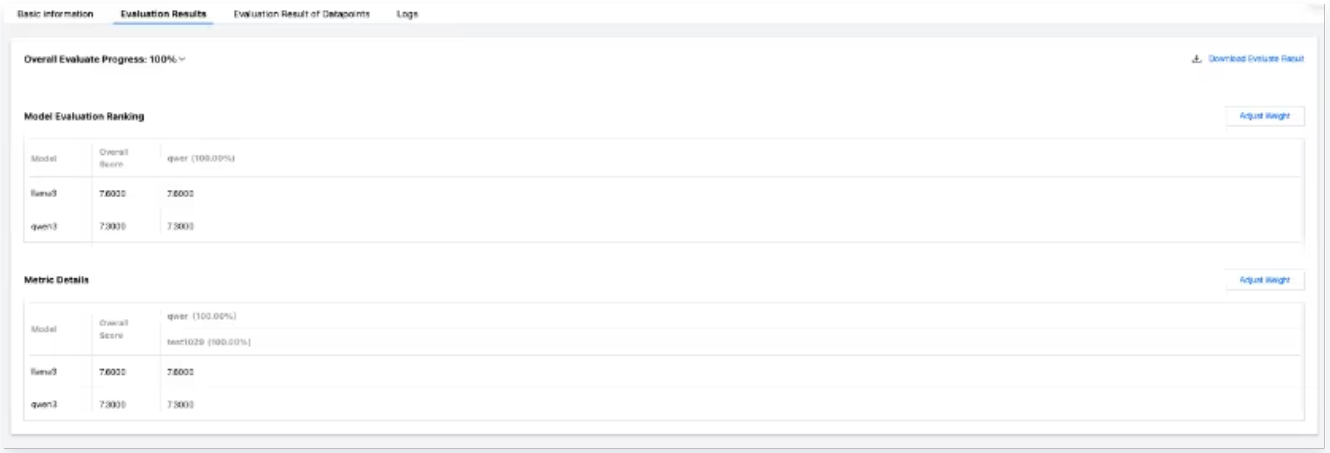
- During the evaluation process, click **View Progress** to view the detailed progress:

©2013-2026 Tencent Cloud. All rights reserved.



- **Overall evaluation results**

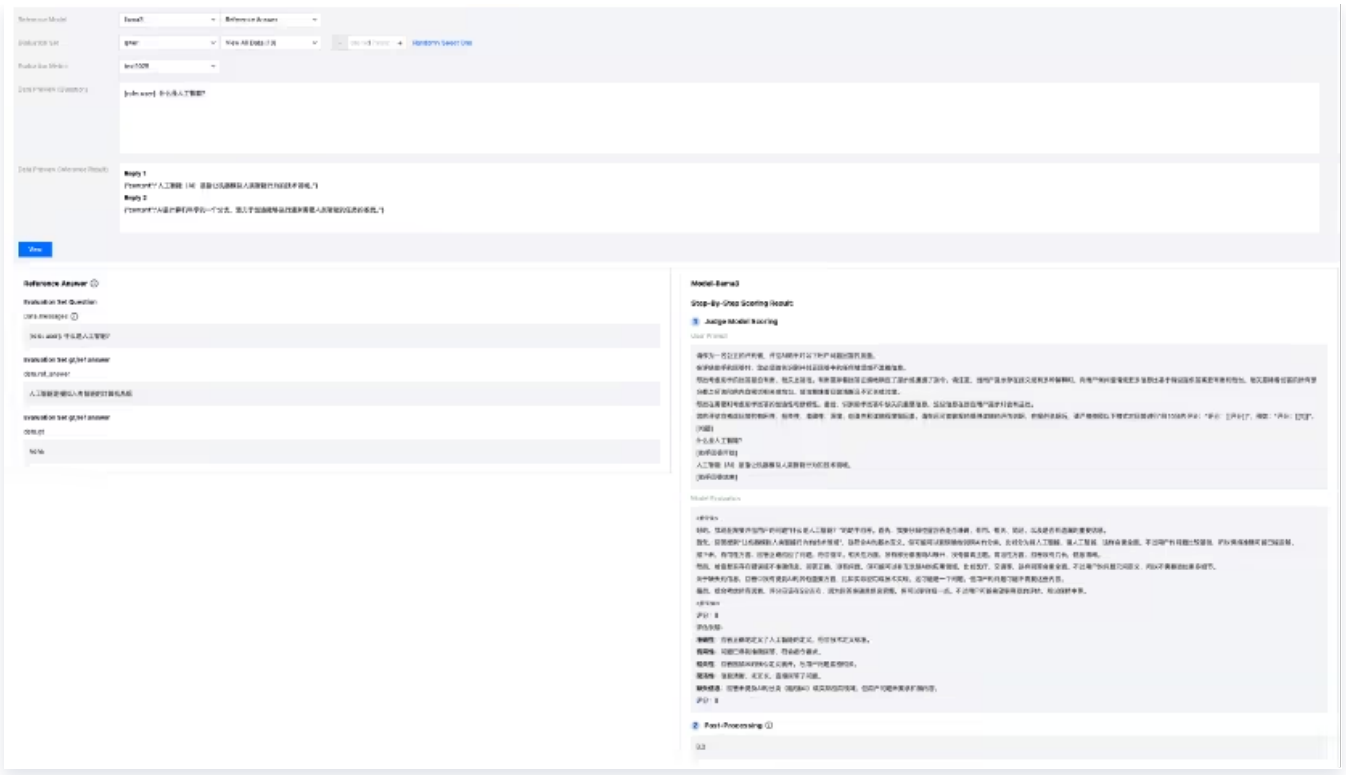
  After the evaluation is completed, you can click the Overall Evaluation Results tab to view the detailed progress of the overall evaluation and its final evaluation results. You can view the comprehensive scores and rankings of the models, as well as detailed metrics for each model. You can also click **Adjust Weights** to adjust the weights of evaluation sets and metrics.



- **Display of a single result**

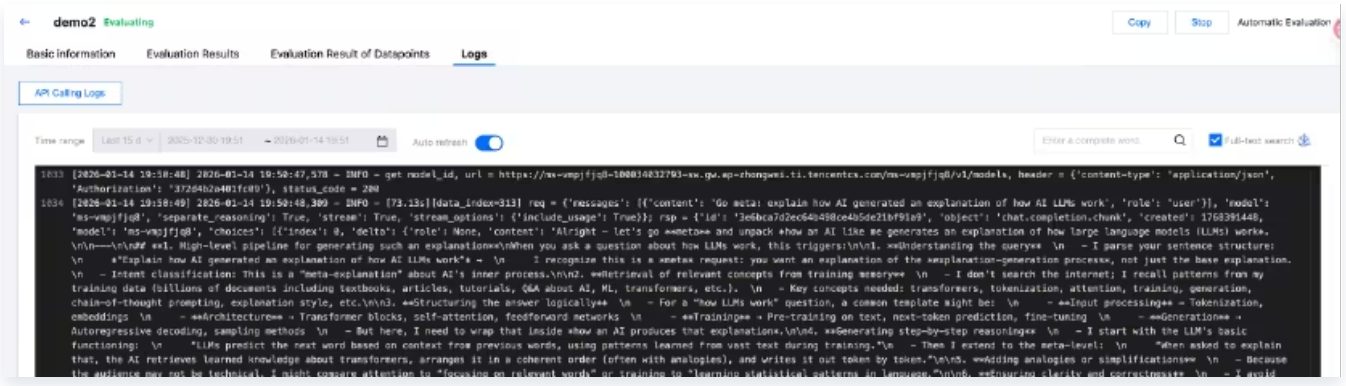
  You can view scoring results for individual data items in steps. You can select a reference model to compare the responses and scoring effects of the model to be evaluated and the reference answers.

©2013-2026 Tencent Cloud. All rights reserved.



- **Log viewing**

  You can view evaluation logs.



2. Custom Evaluation

2.1 Configuring a Basic Task

Parameter	Description
Task Name	Name of an automated evaluation task. You can enter it based on the rules according to the interface prompts.
Remarks	Add remarks to a task as needed.
Region	Services under the same account are isolated by region. The value of the Region field is automatically entered based on the region you selected on the service list

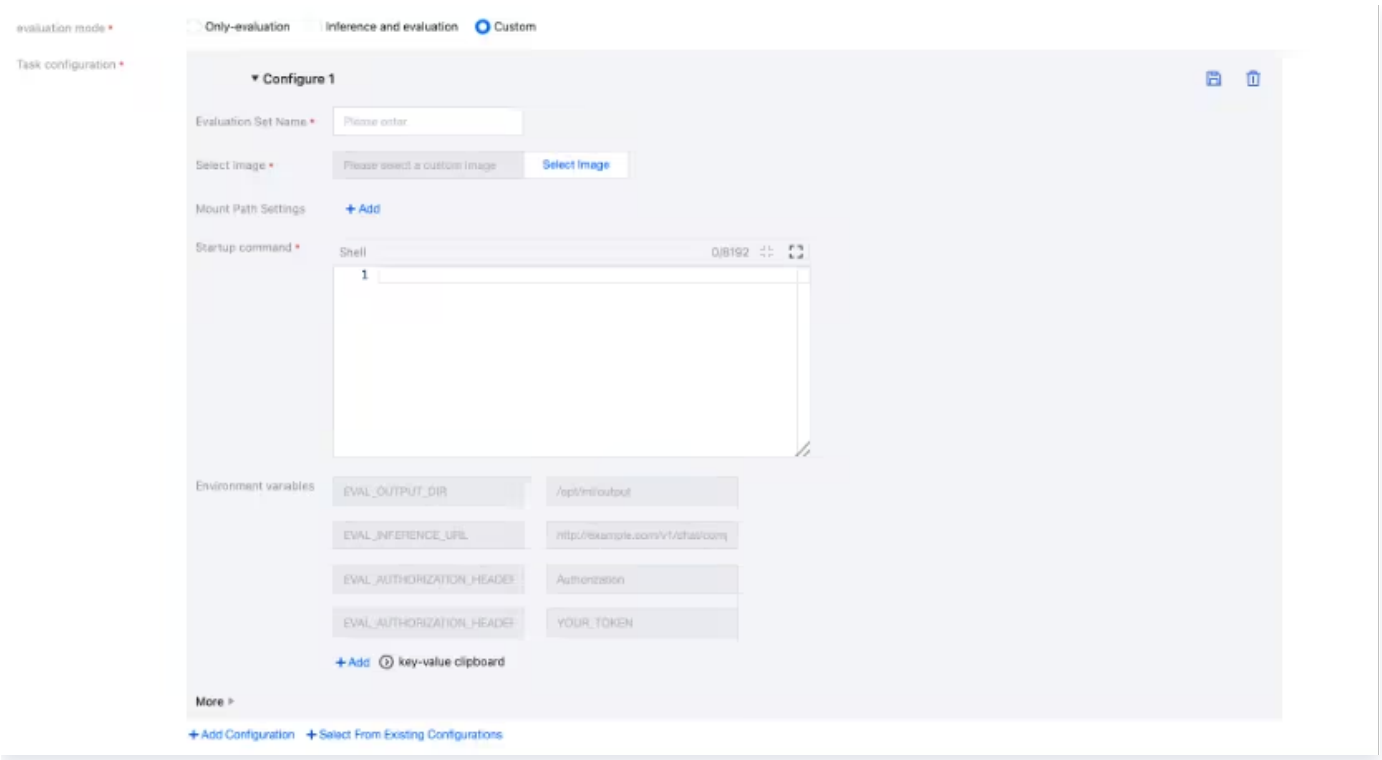
©2013-2026 Tencent Cloud. All rights reserved.

	page.
Tag	It is used for permission isolation between evaluation tasks. Meanwhile, once task tags are configured, the platform will automatically add them to the evaluation configuration.
Billing	• **Billing mode:** You can select the pay-as-you-go mode or yearly/monthly subscription (resource group) mode only when the evaluation mode is inference and evaluation: ○ In the pay-as-you-go mode, you do not need to purchase a resource group in advance. Fees are charged based on the CVM instance specifications on which the service depends. When the service is started, fees for the first two hours are frozen. After that, fees are charged hourly based on the number of running instances. ○ In the yearly/monthly subscription (resource group) mode, you can use the resource group deployment service purchased from the Resource Group Management module. Since computing resource fees are already paid when the resource group is purchased, no fees need to be charged when the service is started. • **Resource group**: If you select the yearly/monthly subscription (resource group) mode, you can select a resource group from the Resource Group Management module. ○ Task resource request: Different tasks require different resource configurations. When multiple configurations exist, it is recommended to specify the maximum resource requirements. ○ Model resource request: When the selected model to be evaluated is a model, you need to specify the model deployment resources according to the actual situation. When multiple models exist, it is recommended to specify the maximum configuration resources.

2.2 Configuring a Task

Evaluation sets, custom images, storage mounts, and other elements can be combined into a task configuration. Each configuration includes evaluation sets, image selection, version selection, mount path settings, startup commands, parameter settings, and environment variables. By selecting images and versions only or selecting images and versions and then configuring mount paths, you can implement evaluations using images only or using images and mount paths.

©2013-2026 Tencent Cloud. All rights reserved.



Below are the use instructions on task configuration:

- **Build a custom image containing evaluation sets and processing logic.**

You can configure and package the basic environment into the image according to the README of the relevant open-source data sets.

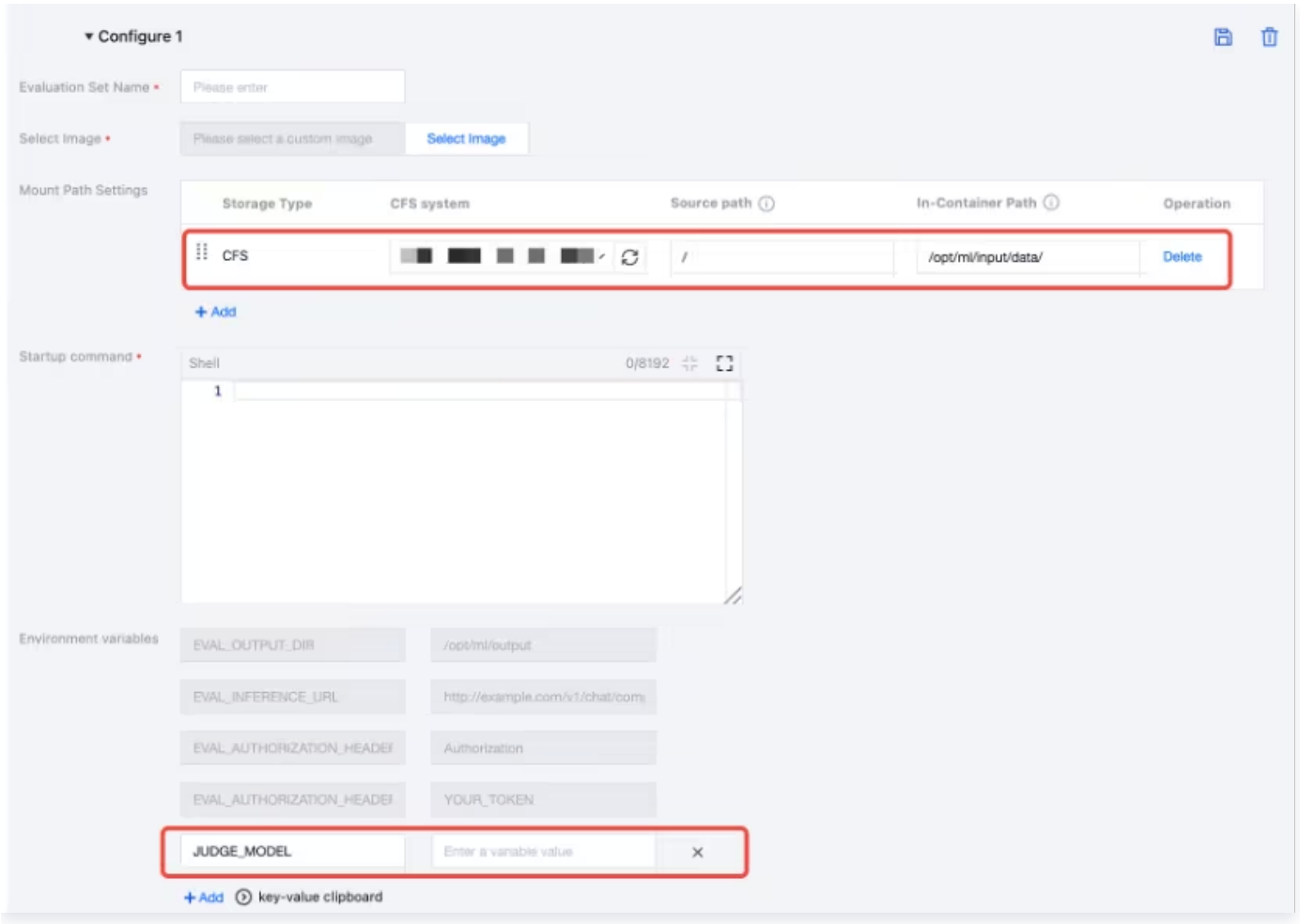
- **Platform environment variables.**

The platform will automatically inject the following 4 environment variables into your image, as follows:

Environment Variable	Description	Remarks	Value
EVAL_OUTPUT_DIR	Result output directory	The evaluation results must be saved to this directory in the image for result display/visual comparison.	/opt/ml/output
EVAL_INFERENCE_URL	Model API address	Since a task configuration may evaluate multiple model services, the platform passes the call information of model services through this environment variable.	https://api.openai.com/v1/chat/completions
EVAL_AUTHORIZATION_HEADER_KEY	Authentication header name		Authorization
EVAL_AUTHORIZATION_HEADER_VALUE	Authentication header value		Bearer sk-xxx...

You can also inject configurations such as judge models and inference results into the execution environment via environment variables or storage mounts to facilitate calls, as shown in the following example:

©2013-2026 Tencent Cloud. All rights reserved.



2.3 Configuring a Model/Service to Be Evaluated

Call according to the relevant environment variables in section 2.2. The platform will pass information about the selected online services/third-party URLs during execution. You need to select the model or service to be evaluated to generate inference results for evaluation. Sources for models to be evaluated include built-in LLMs/training tasks/CFS/GooseFSx/data sources/resource groups, while sources for model services include Online Service/Specify Service Addresses. When users configure models/services to be evaluated, the parameter setting feature is provided.

- Select a model:
  - If you select a built-in LLM, you can quickly enable an evaluation.
  - If you select a model from training tasks, you can select a training task in the current region or the checkpoint of the task.
  - If you select a model from data sources, you need to choose **Platform Management > Data Source Management** to create a data source. Note: The data source mount permissions are divided into read-only mount and read-write mount. For a data source for which you need to output results, you can set its mount permission to read-write.
  - If you select a model from CFS or GooseFSx, you need to select the CFS instance or GooseFSx instance from the drop-down list and enter the directory to be mounted by the

©2013-2026 Tencent Cloud. All rights reserved.

platform.

- Select a service:
  - If you select a service from Online Services, you need to select the online service name. To ensure that the evaluation can be initiated successfully, make sure that the service is running normally. In addition, you need to select the corresponding authentication information.
  - If you choose to enter a service address, you need to enter the complete service call URL along with the corresponding authentication information. If you use this method to select a service, you can check whether the service can run normally through the connectivity test.
- Configure parameters: You can configure inference hyperparameters, startup parameters, and performance parameters.
  - Configure inference hyperparameters as follows:

    repetition_penalty: controls the repetition penalty.
    max_tokens: controls the maximum length of the output text.
    temperature: A higher temperature makes outputs more random; a lower temperature makes outputs more focused and deterministic.
    top_p and top_k: control the diversity of the output text. Higher values produce more diverse outputs. It is recommended to configure only 1 of the temperature, top_p, and top_k parameters.
    do_sample: specifies the sampling method for model inference. When this parameter is set to true, the sample method is used; when this parameter is set to false, the greedy search method is used, and the top_p, top_k, temperature, and repetition_penalty parameters do not take effect.
  - Configure startup parameters: more details in Service Deployment Parameter Configuration Guide. MAX_MODEL_LEN is the default parameter configured on the platform, which specifies the maximum number of tokens a model can process in a single inference operation. Its default value is 8192 on the platform. If you set this parameter to a very high value upon startup, GPU out-of-memory or performance degradation issues may occur. You can adjust this value appropriately based on task requirements.
  - Configure performance parameters: MAX_CONCURRENCY and MAX_RETRY_PER_QUERY are the default parameters configured on the platform.

    MAX_CONCURRENCY indicates the maximum number of requests that can be sent to a model simultaneously during evaluation. Setting this value too low may cause the throughput of a model to decrease and lead to a long evaluation time, while setting this value too high may cause GPU out-of-memory or request timeout issues. Its default value is 24 on the platform. You can adjust this value appropriately based on task requirements.
    MAX_RETRY_PER_QUERY indicates the maximum number of retries for each piece of data when an exception occurs in requesting the inference service, such as a request

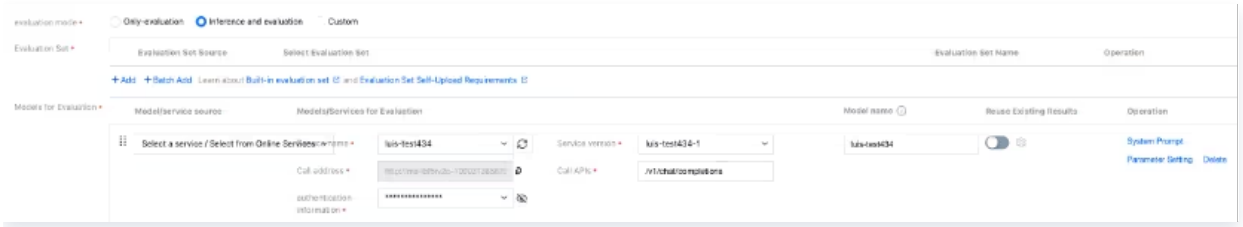
©2013-2026 Tencent Cloud. All rights reserved.

timeout or a network failure. If the value is 0, no retry is performed (default value: 0). You can adjust this value appropriately based on task requirements.

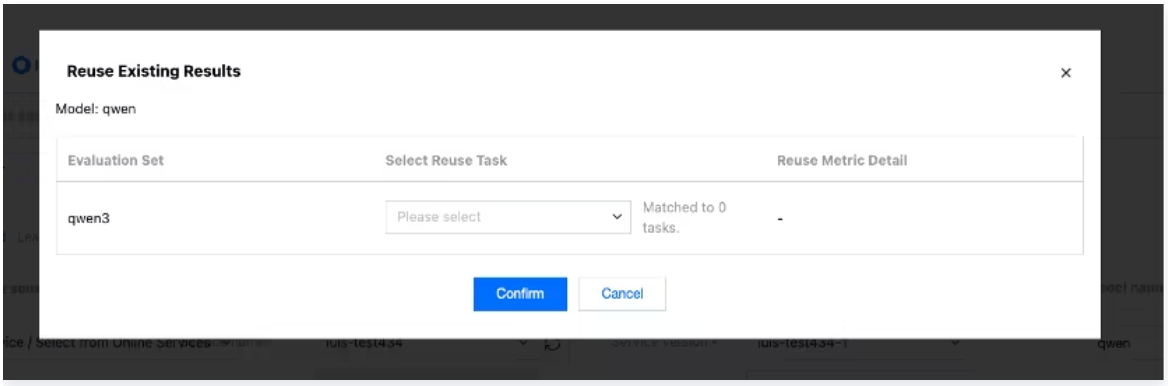
- You can reuse previous evaluation results.

> **Note:**
> Ensure that the path for outputting metrics to the platform is configured in the custom evaluation. Otherwise, task metrics cannot be obtained.

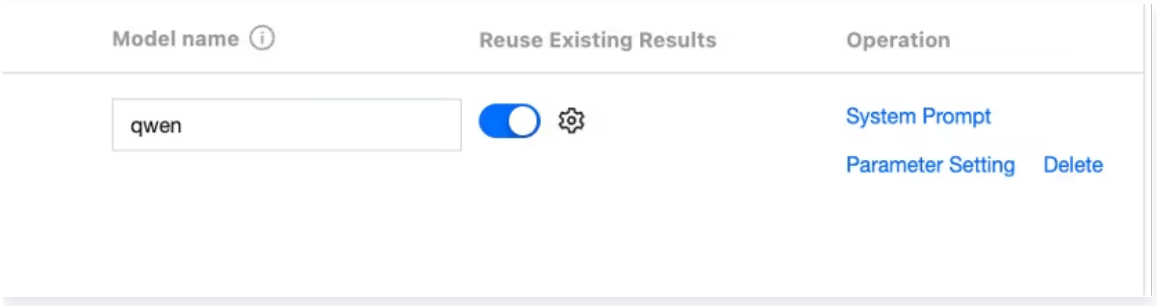
When you select a model to be evaluated, you can select the evaluation results of a previous evaluation task with the same model name and the same evaluation set name.



Enable the reuse switch and select the evaluation tasks to be reused.



The results can be directly reused without performing inference and scoring again.



2.4 Viewing Evaluation Results

> **Note:**
> Execute the following steps to view evaluation results. After completing the following steps, you can view metrics in the overall evaluation results. Before executing the steps,

©2013-2026 Tencent Cloud. All rights reserved.

> you can only view evaluation results via custom image logs, and you cannot select the evaluation results for this evaluation set on relevant models in the visual comparison.

To view the evaluation results for this data set on relevant models on the details page of custom evaluation results, you need to store the specific evaluation results in the platform path `${EVAL_OUTPUT_DIR}`. The name of the file is metrics.json, and its format is as follows:

```
{
  "metrics": [
    {
      "name": "accuracy",
      "value": 0.85
    },
    {
      "name": "f1_score",
      "value": 0.78
    }
  ]
}
```

You can write the file in the custom image via the following code:

```
import json
import os

# Obtain the output directory.
output_dir = os.getenv('EVAL_OUTPUT_DIR')

# Construct the result data.
result = {
    "metrics": [
        {"name": "accuracy", "value": 0.85},
        {"name": "f1_score", "value": 0.78}
    ]
}

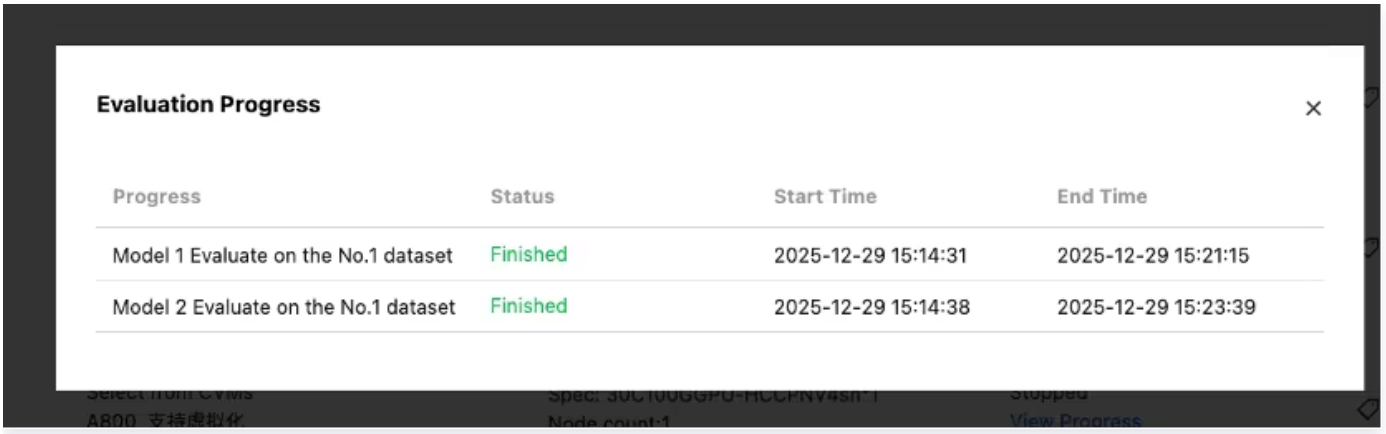
# Write the result file.
result_file = os.path.join(output_dir, 'metrics.json')
with open(result_file, 'w', encoding='utf-8') as f:
    json.dump(result, f, indent=2, ensure_ascii=False)
```

©2013-2026 Tencent Cloud. All rights reserved.

Click the Overall Evaluation Results tab to view the detailed progress and final results of the overall evaluation. Viewing individual evaluation results is currently not supported.

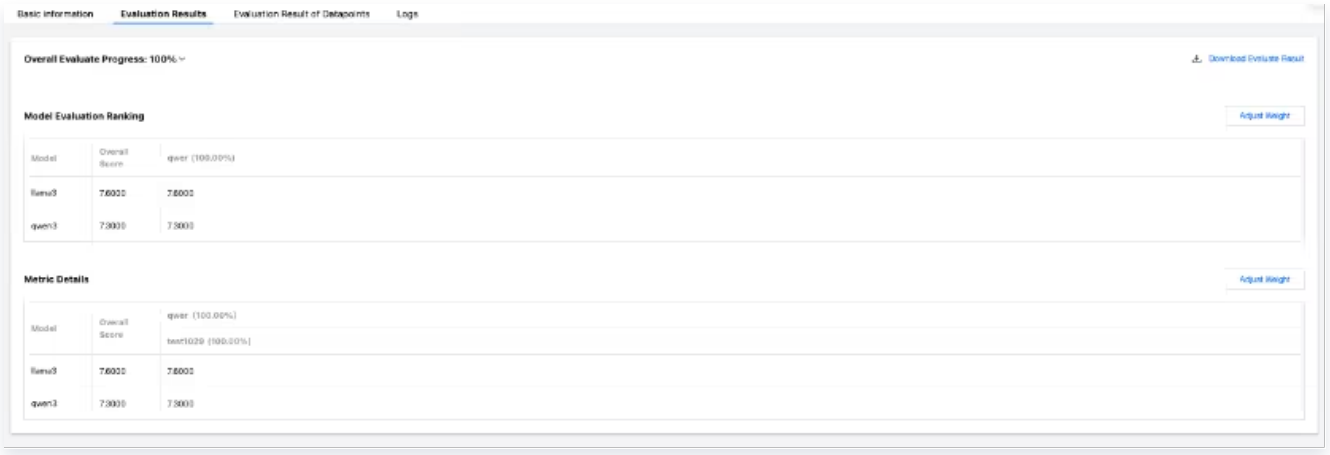
- **Overall evaluation progress**

> **Note:**
> In the custom evaluation mode, the platform cannot obtain the number of evaluation sets. The calculation formula for the current progress is: Progress = Completed evaluation tasks (Models x Evaluated evaluation sets)/Total evaluation tasks (Models x Total evaluation sets). The evaluation progress can remain unchanged for an extended period due to its large granularity, but this does not mean that the evaluation has paused or failed. You can view the progress via logs.



- **Overall evaluation results**

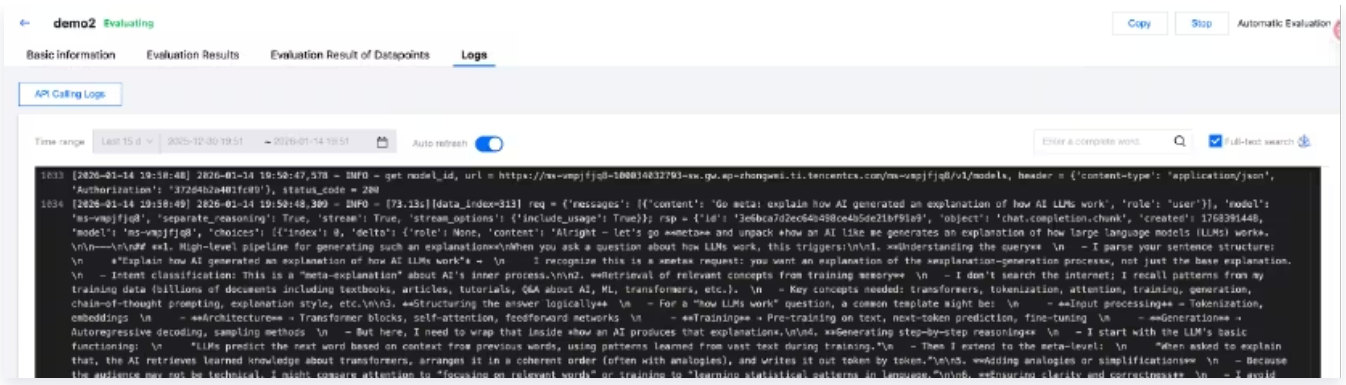
You can view the comprehensive scores and rankings of the models, as well as detailed metrics for each model. You can also click **Adjust Weights** to adjust the weights of evaluation sets and metrics.



- **Log viewing**

  You can view evaluation logs.

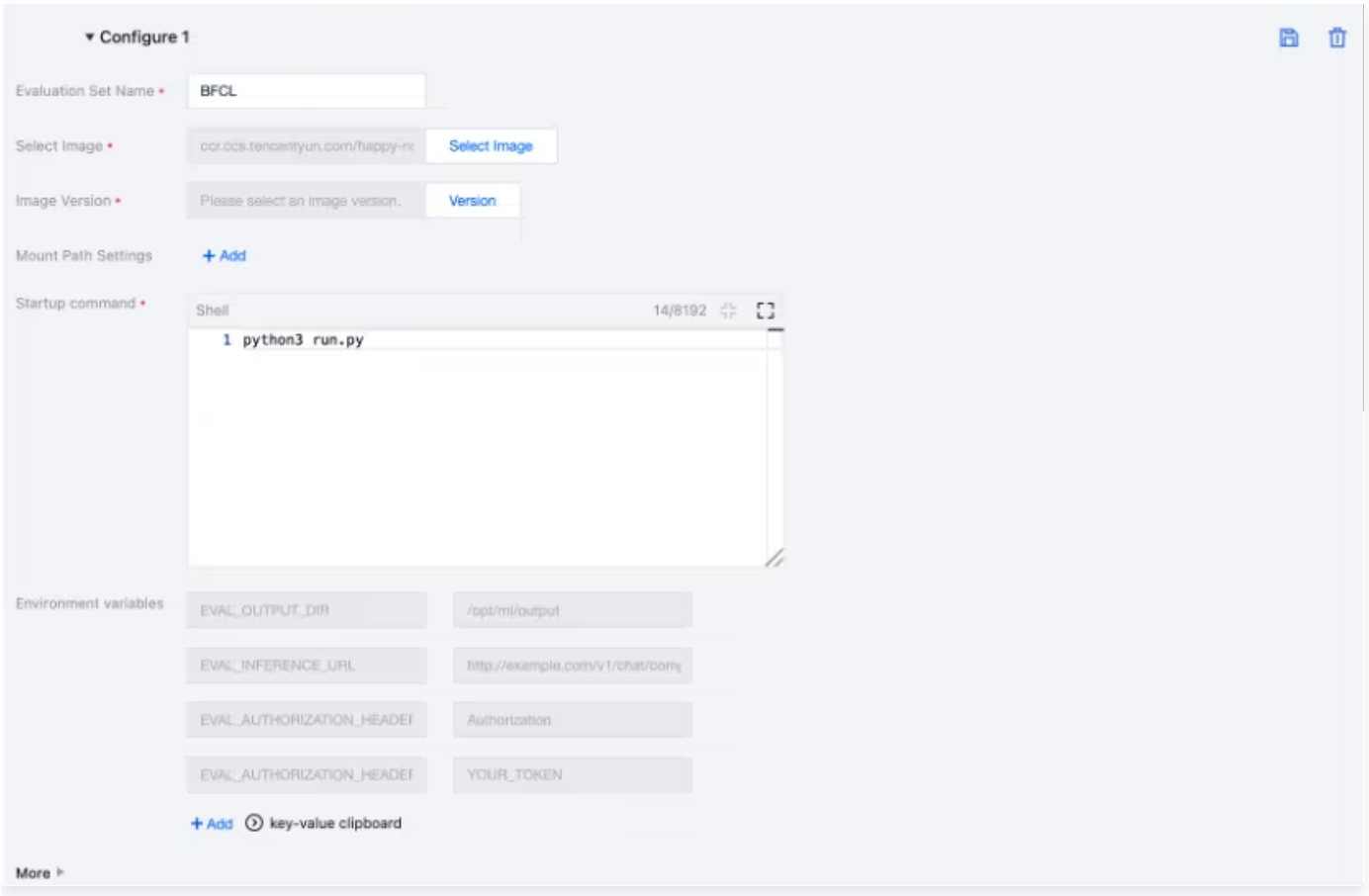
©2013-2026 Tencent Cloud. All rights reserved.



2.5 Custom Evaluation Configuration Example (BFCL Evaluation Set)

The task configuration for each evaluation set is executed serially on each model service.

- Background: Evaluation images are customized to perform evaluations by mounting BFCL evaluation sets.
- Task configuration: You need to customize evaluation images, storage mounts, environment variables, and startup commands of the BFCL evaluation set.



©2013–2026 Tencent Cloud. All rights reserved.

Manual Evaluation

Last updated: 2026-03-12 10:24:28

Manual evaluation is to evaluate the output effectiveness of an LLM and provides a manually-annotated scoring mode to objectively evaluate the model effect.

Prerequisites

Preparing a Custom Evaluation Set

For details, see Evaluation Set Format Requirements.

After the evaluation set is prepared, you need to select the source (Cloud File Storage (CFS)/GooseFSx/data sources) of the evaluation set and its path when you create a task. CFS is used as an example. To facilitate task creation, during the preparation of the evaluation set, you need to mount your CFS instance to dev machines to obtain the path required during evaluation. The CFS instance is used as follows: To facilitate task creation, during the preparation of the evaluation set, you need to mount your CFS instance to dev machines to obtain the path required during evaluation. The CFS instance is used as follows:

1. Prepare the CFS instance. You can mount your CFS instance and start dev machines. Assume that the CFS instance is already prepared and that /data1 represents the root path of the CFS instance mounted locally.
2. You can create a local folder for the subjective data set you want to use for evaluation as needed.
3. For example, you can place the required subjective data set test.csv under the `/data1/test_data` directory:

```
cd /data1
mkdir -p test_data
cp userdata/test.csv test_data
```

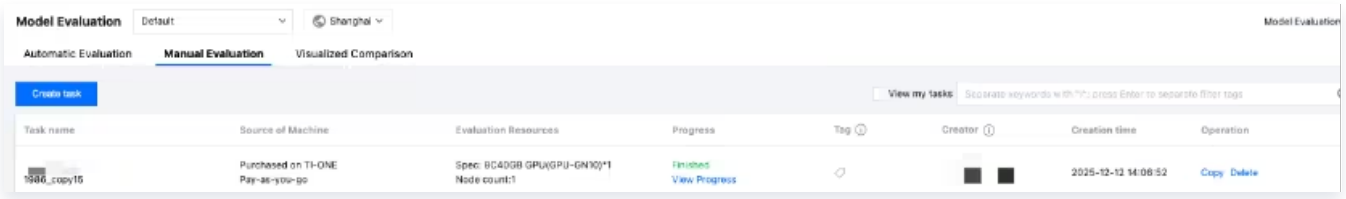
4. During evaluation, enter the directory where your data set is located on the CFS instance, for example, `/data1/test_data.`

> **Note:**
> If multiple evaluation set files exist under the entered directory, model evaluation will evaluate every evaluation set file.

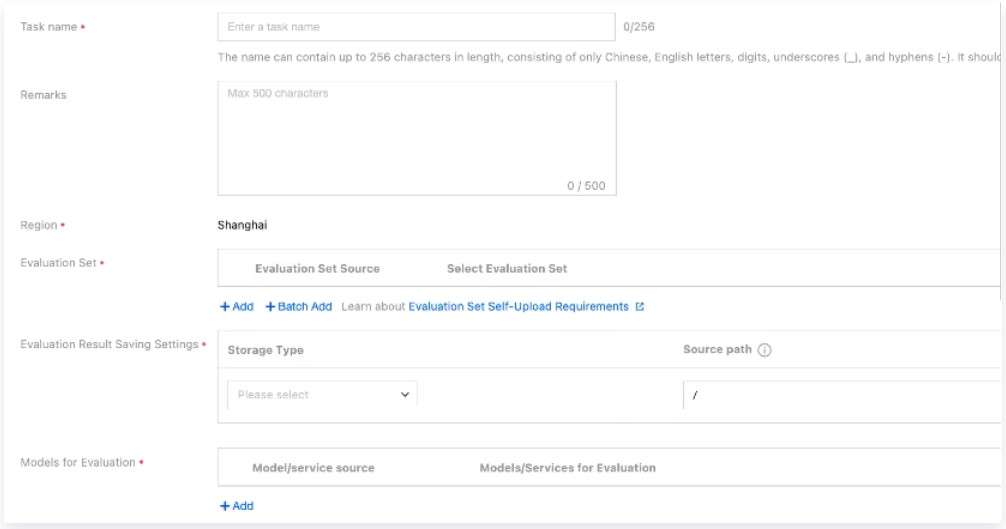
Operation Steps

©2013-2026 Tencent Cloud. All rights reserved.

1. Log in to the TI-ONE Platform, choose **Model Services > Model Evaluation** in the left sidebar, and then click the **Manual Evaluation** tab to go to the task list page.



2. Click **New Task** to go to the creation page.



The following table describes the required information:

Parameter	Description
Task Name	Name of a manual evaluation task. You can enter it based on the rules according to the interface prompts.
Remarks	Add remarks to a task as needed.
Region	Services under the same account are isolated by region. The value of the Region field is automatically entered based on the region you selected on the service list page.
Evaluation Set	• You can select the directory where the CFS instance and evaluation data set are located (CFS/GooseFSx/data sources). ○ If you select a data source, you need to choose **Platform Management > Data Source Management** to create a data source. Note: The data source mount permissions are divided into read-only and read-write. For a data source for which you need to output training results, you can set its mount permission to read-write. ○ If you select CFS or GooseFSx, you can select the directory where the CFS instance and the evaluation data set are located. Only the

©2013-2026 Tencent Cloud. All rights reserved.

	JSONL and CSV formats are supported, and the content contains 2 columns, namely, prompt and answer respectively. • You can select a built-in evaluation set and enable quick evaluation with one click.
Save Evaluation Results	You can set the storage path to the directory where the CFS instance and the evaluation data set are located (CFS/GooseFSx/data sources). • If you select a data source, you need to choose **Platform Management > Data Source Management** to create a data source. Note: The data source mount permissions are divided into read-only and read-write. For a data source for which you need to output training results, you can set its mount permission to read-write. • If you select CFS or GooseFSx, you need to select the CFS instance or GooseFSx instance from the drop-down list and enter the data source directory to be mounted by the platform.
Select the Model to Be Evaluated	Two sources of models to be evaluated are supported: • Select Model: ○ Select a built-in LLM. ○ Select a model from training tasks. For example, you can select a training task in the current region and then the checkpoint of the task. ○ Select a model from CFS, GooseFSx, or data sources: ○ If you select a data source, you need to choose **Platform Management > Data Source Management** to create a data source. Note: The data source mount permissions are divided into read-only and read-write. For a data source for which you need to output training results, you can set its mount permission to read-write. ○ If you select CFS or GooseFSx, you need to select the CFS instance or GooseFSx instance from the drop-down list and enter the data source directory to be mounted by the platform. • Select Service: ○ Select a service from the Online Services module of TI-ONE. ○ Enter the address of a third-party service for evaluation. You can configure parameters, including inference hyperparameters, startup parameters, and performance parameters. • Configure inference hyperparameters as follows: ○ repetition_penalty: controls the repetition penalty. ○ max_tokens: controls the maximum length of the output text. ○ temperature: A higher temperature makes outputs more random; a lower temperature makes outputs more focused and deterministic.

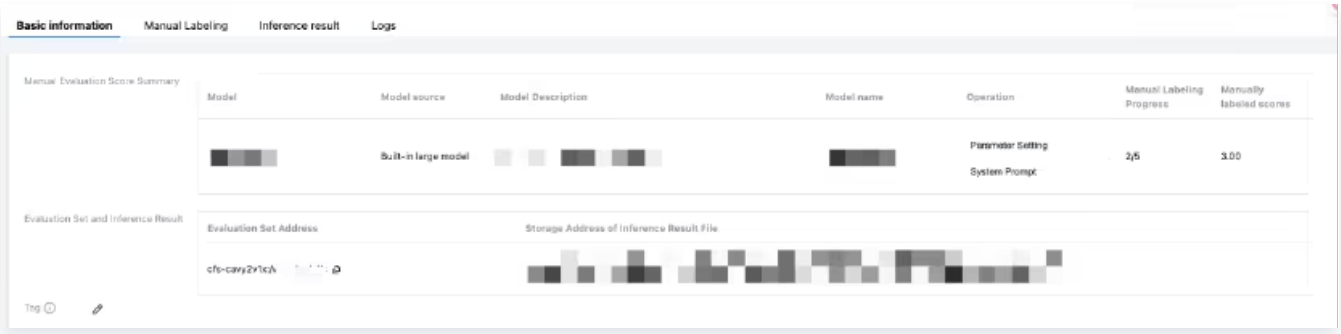
©2013-2026 Tencent Cloud. All rights reserved.

	○ top_p and top_k: control the diversity of the output text. Higher values produce more diverse outputs. It is recommended to configure only 1 of the temperature, top_p, and top_k parameters. ○ do_sample: specifies the sampling method for model inference. When this parameter is set to true, the sample method is used; when this parameter is set to false, the greedy search method is used, and the top_p, top_k, temperature, and repetition_penalty parameters do not take effect. • Configure startup parameters: For details, see Service Deployment Parameter Configuration Guide. MAX_MODEL_LEN is the default parameter of the platform, which specifies the maximum number of tokens a model can process in a single inference operation. Its default value is 8192 on the platform. If you set this parameter to a very high value upon startup, GPU out-of-memory or performance degradation issues may occur. You can adjust this value appropriately based on task requirements. • Configure performance parameters: MAX_CONCURRENCY and MAX_RETRY_PER_QUERY are the default parameters of the platform. ○ MAX_CONCURRENCY indicates the maximum number of requests that can be sent to a model simultaneously during evaluation. Setting this value too low may cause the throughput of a model to decrease and lead to a long evaluation time, while setting this value too high may cause GPU out-of-memory or request timeout issues. Its default value is 24 on the platform. You can adjust this value appropriately based on task requirements. ○ MAX_RETRY_PER_QUERY indicates the maximum number of retries for each piece of data when an exception occurs in requesting the inference service, such as the request timeout or network failure. If the value is 0, no retry is performed (default value: 0). You can adjust this value appropriately based on task requirements.
Billing Mode	You can select pay-as-you-go or yearly/monthly subscription (resource group): • (A) In pay-as-you-go mode, you do not need to purchase a resource group in advance. Fees are charged based on the CVM instance specifications on which the service depends. When the service is started, fees for the first two hours are frozen. After that, fees are charged hourly based on the number of running instances. • (B) In yearly/monthly subscription (resource group) mode, you can use the resource group deployment service purchased from the Resource Group Management module, and computing resource fees are already paid when the resource group is purchased. Therefore, no fees need to be charged when the service is started.

©2013-2026 Tencent Cloud. All rights reserved.

Resource Group	If you select the yearly/monthly subscription (resource group) mode, you can select a resource group from the Resource Group Management module.

3. After you enter the corresponding information and create a manual evaluation task, the following information will be displayed on the task list page: Task Name, CVM Instance Source, Evaluation Resources, Status, Progress, Tag, Creator, Creation Time, and Operations (Stop, Restart, Delete, and Copy).
4. Click **Inference Progress** to download and view the evaluation result set under the current progress.
5. After the manual evaluation task is completed, you can click the task name to go to the task details page. On this page, you can view the basic information, perform manual annotation, and view the evaluation results and logs.



©2013-2026 Tencent Cloud. All rights reserved.

Evaluation-based Visual Comparison

Last updated: 2026-01-15 14:24:49

Overview

Evaluation-based visual comparison provides an intuitive display of metric comparison effects for different models through radar charts. It allows direct, quick selection and comparison based on evaluation metrics from the completed automated evaluation and manual evaluation. It also provides an effect comparison for different models on the same aggregated metrics after the aggregated metrics are configured. The detailed description of its feature is as follows:

- Method 1 **comparison of evaluation metrics**: You can directly perform horizontal comparison on multiple metrics of different models after automated evaluation and manual evaluation are completed.
- Method 2 **comparison of aggregated metrics**: After aggregated metrics are configured, you can select the automated evaluation task to perform horizontal comparison on the same aggregated metrics of different models.

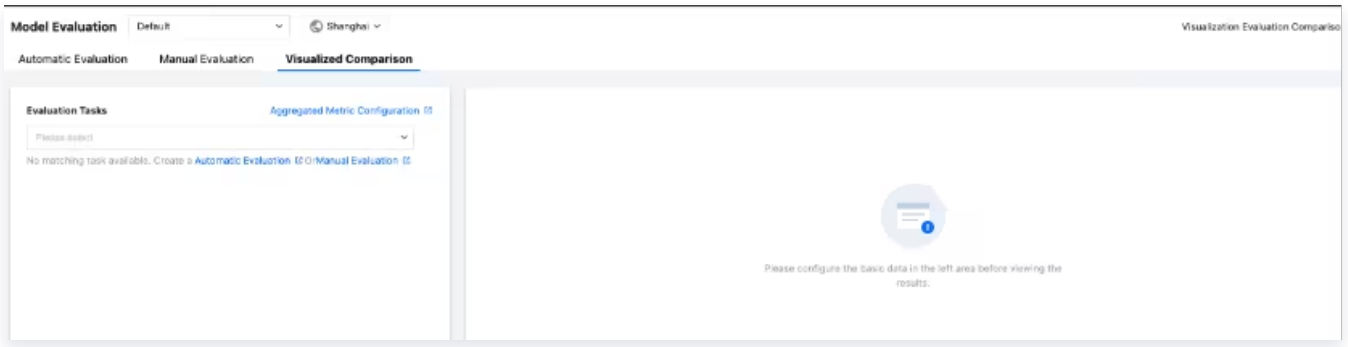
Method 1: Comparison of Evaluation Metrics

Prerequisites

You have completed model evaluation through Automated Evaluation and Manual Evaluation. The status of the automated evaluation task is **Completed**, and the status of the manual evaluation task is **Scoring Completed**.

Configuring the Visual Comparison Chart

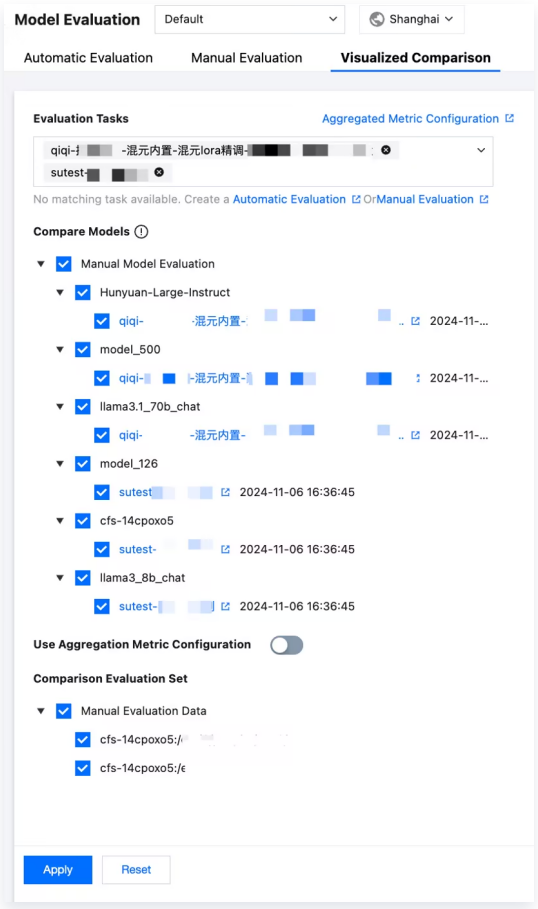
1. Select **evaluation tasks**. You can select the automated evaluation tasks in **Completed** status and the manual evaluation tasks in the **Scoring Completed** status. A maximum of 10 tasks can be selected.



2. After evaluation tasks are selected, the evaluation data sets and models used by the selected tasks will be automatically displayed. Both data sets and models are selected by default, but

©2013-2026 Tencent Cloud. All rights reserved.

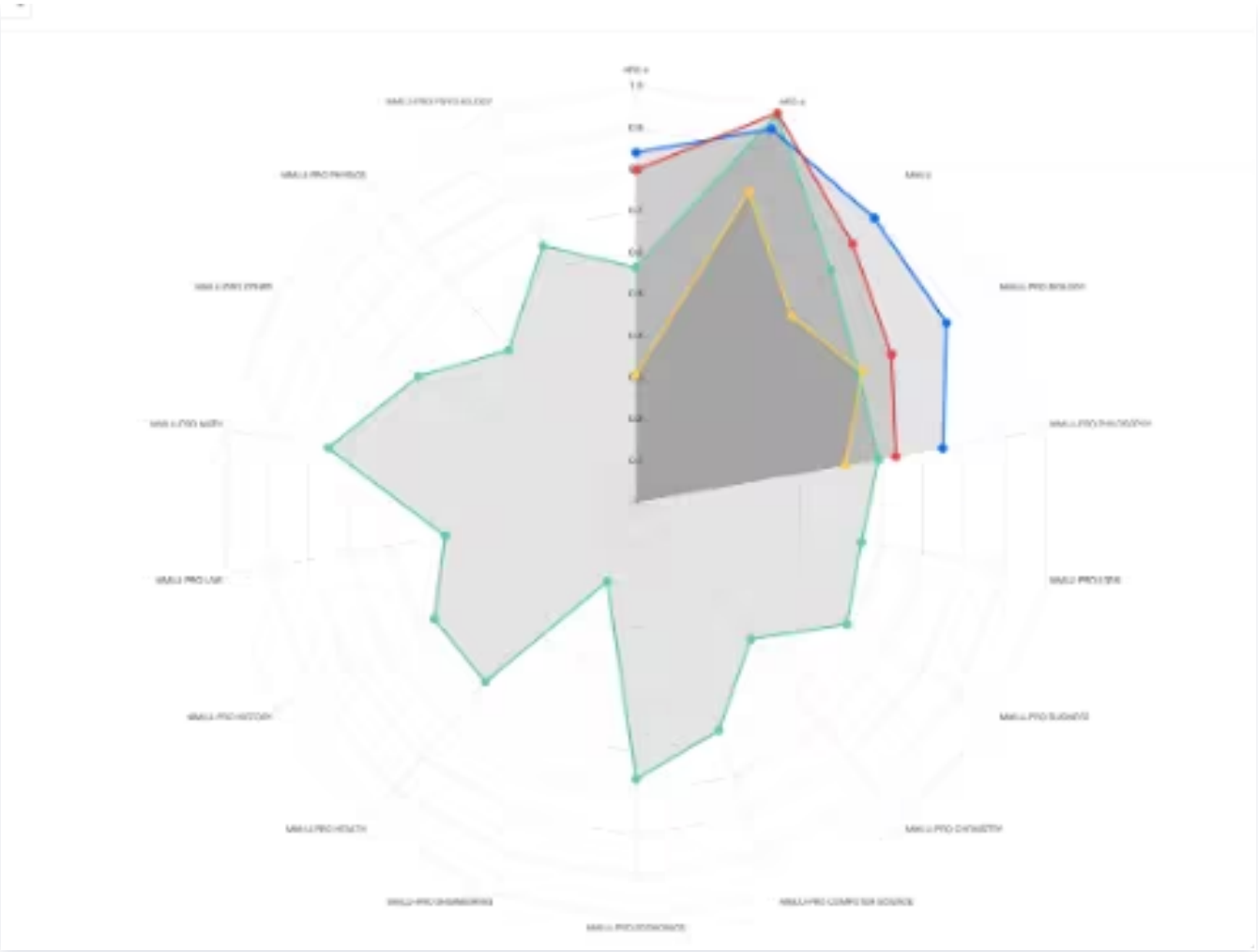
you can deselect them as needed. Evaluation tasks related to these models will be listed according to the latest evaluation time, and you can select related evaluation tasks, as shown in the following figure:



> **Note:**
> You can select multiple evaluation tasks. If you select multiple evaluation tasks that contain the same model and evaluation set, the values for the same model on the same evaluation set will be averaged.

3. Click **Apply**. The selected evaluation tasks will be displayed on the right side, and a comparison of evaluation metrics will be displayed through a radar chart based on the selected model and data set, as shown in the following figure:

©2013-2026 Tencent Cloud. All rights reserved.



Method 2: Comparison of Aggregated Metrics

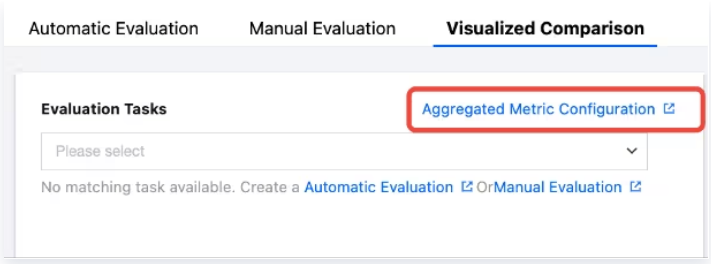
Prerequisites

You have configured aggregated metrics, and the status of the automated evaluation task is **Completed**.

Configuring Aggregated Metrics

Overview:
Aggregated metrics are used to measure specific capabilities of a model. Generally, aggregation is carried out using different metrics from different evaluation sets.
For example, if you want to evaluate the comprehensive general capabilities of a model across multiple disciplines, you can configure aggregated metrics in various dimensions such as low difficulty, high difficulty, interdisciplinary research, biology, and philosophy. In the subsequent comparative evaluation, you can obtain comparison results on the aggregated metrics directly by selecting this configuration.
Usage method:
1. Click **Aggregated Metrics Configuration** to go to the aggregated metric configuration page.

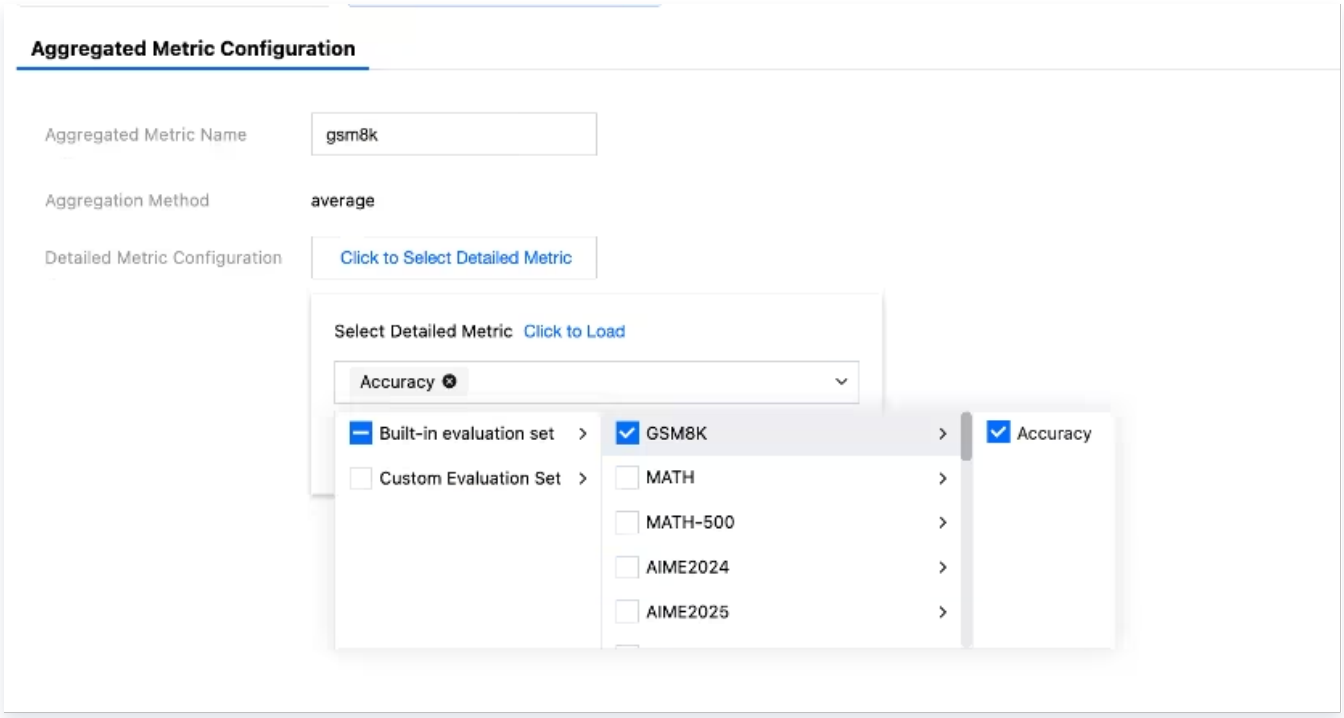
©2013-2026 Tencent Cloud. All rights reserved.



2. Click **+Aggregated Metrics** to add an aggregated metric card and configure the aggregated metric name.

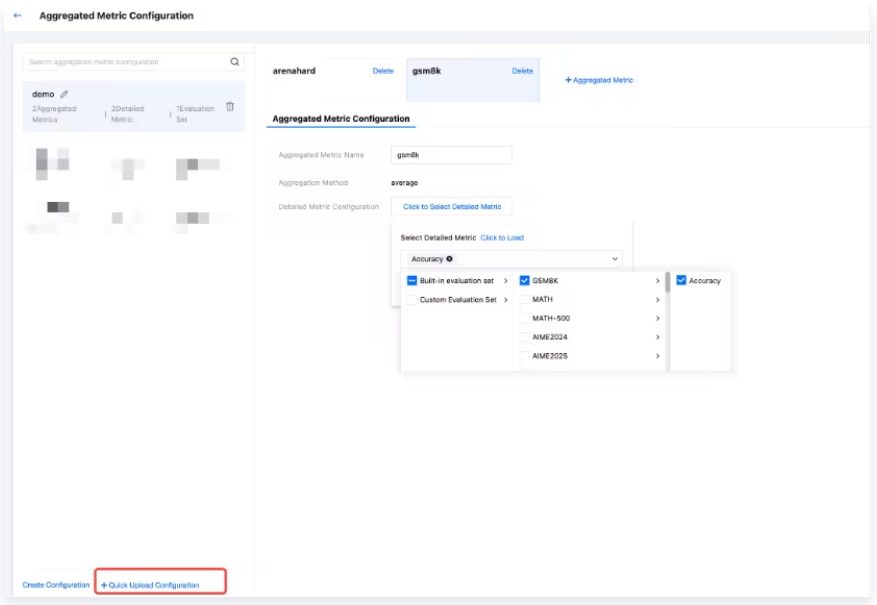


3. Click **Select Detailed Metrics**, and select the corresponding evaluation set and metric based on the corresponding aggregated metric.



4. **Quick configuration** is supported. You can rapidly define and manage the aggregation rules of evaluation metrics for multiple data sets through a YAML configuration file.

©2013-2026 Tencent Cloud. All rights reserved.



Configuration file format: Only the YAML format is supported. Supported file extensions: .yaml and .yml.
Basic structure: A configuration file for aggregated metrics can contain multiple aggregated metrics, and you can configure them separately.

```yaml
- name: test                    # Name of an aggregated metric.
  method: average               # Aggregation method.
  metrics:                      # Metric configuration list.
    - data_name: "MMLU"         # Data set name.
      metric_names:             # List of metric names.
        - "accuracy"
        - "pass@1"
    - data_name: "aime2025"
      metric_names:
        - "fluency"
        - "correlation"
```

Configuration parameter description

Parameter Name	Parameter Description	Parameter Example
name	Name of an aggregated metric.	test
method	Aggregation calculation method. Currently, average (average value) is supported.	average

©2013-2026 Tencent Cloud. All rights reserved.

metrics	Metric configuration array. It is used to define the data sets and metrics to be aggregated.	–
data_name	Data set name. It corresponds to the data set of an evaluation task.	"MMLU"
metric_names	Metric name array. It is used to list the metrics to be aggregated under the data set.	"accuracy"

Usage example

Create aggregated metric configurations and a YAML file (for example, aggregation_config.yaml), and define aggregation rules in the format above:

```yaml
- name: "comprehensive evaluation metric"
  method: average
  metrics:
    - data_name: "MMLU"
      metric_names:
        - "accuracy"
        - "pass@1"
    - data_name: "GSM8K"
      metric_names:
        - "accuracy"
        - "exact_match"
    - data_name: "HumanEval"
      metric_names:
        - "pass@1"
        - "pass@10"
- name: "mathematical inference capability"
  method: average
  metrics:
    - data_name: "GSM8K"
      metric_names:
        - "accuracy"
    - data_name: "MATH"
      metric_names:
        - "accuracy"
        - "step_accuracy"
```

©2013–2026 Tencent Cloud. All rights reserved.

Configuring the Visual Comparison Chart

1. Select **Evaluation Tasks**. A maximum of 10 automated evaluation tasks in the **Completed** status can be selected.



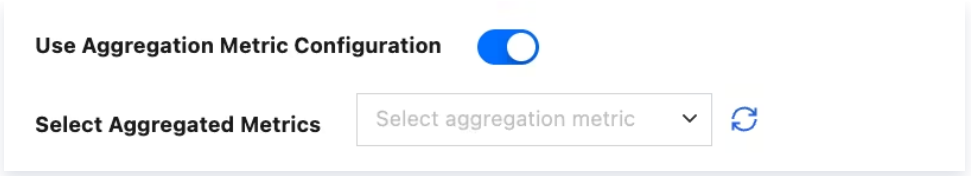
2. After evaluation tasks are selected, the **model** used by the selected tasks will be automatically displayed. The evaluation tasks related to this model will be listed based on the latest evaluation time. You can select related evaluation tasks.



> **Note:**
> You can select multiple evaluation tasks. If multiple evaluation tasks that contain "a model evaluates an evaluation set" are selected, the values of the evaluation metrics of this model on this evaluation set will be averaged.

3. Enable the **Use Aggregated Metric Configuration** switch and select aggregated metric configuration.

©2013–2026 Tencent Cloud. All rights reserved.



The evaluation sets in the aggregated metric configuration will be listed. You can select the evaluation sets to be compared as needed. Note: When an evaluation set is deselected, the aggregated metrics related to this evaluation set will not be involved in the comparison.

4. In addition, you can also select the evaluation result file (in JSONL format) through Cloud File Storage (CFS). During the comparison evaluation, the platform will display comparison results based on the evaluation result file and the selected tasks mentioned above.

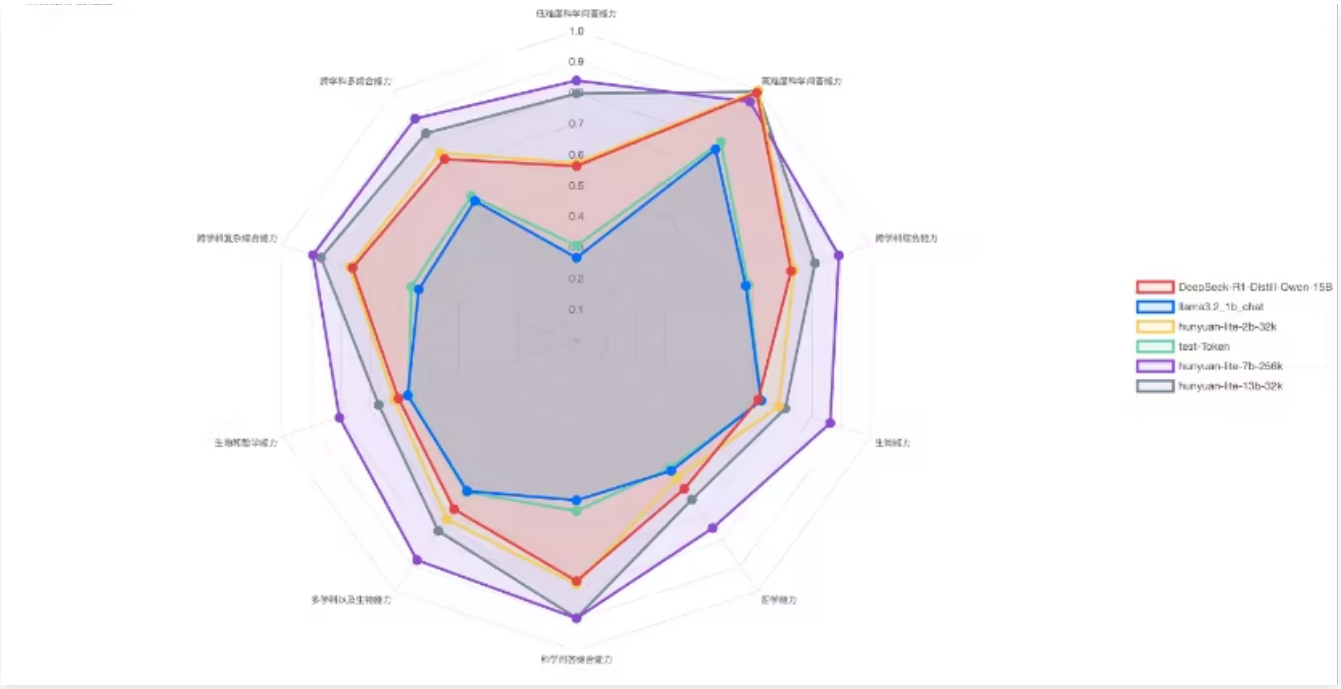
   Description for local evaluation result specifications:

```
{"ModelName":"model1","DatasetName":"AIME2024","DatasetScore":0.83,"Metric
Name":"Accuracy","MetricValue":0.83,"Weight":1}
```

Field Name	Required	Field Description
ModelName	Required	Model name
DatasetName	Required	Evaluation set name
DatasetScore	Optional	Score of the evaluation set
MetricName	Required	Detailed metric name
MetricValue	Required	Detailed metric value
Weight	Required	Weight

5. Click **Apply**. The selected evaluation tasks will be displayed on the right side, and a comparison will be displayed through a radar chart based on the selected model and aggregated metrics, as shown in the following figure.

©2013-2026 Tencent Cloud. All rights reserved.



In addition, the capabilities of each model on aggregated metrics will be **ranked**. You can also find the detailed data of each aggregated metric, as well as the detailed data of each metric on each evaluation set, from the table. The table uses cells with different shades of colors to help you easily and intuitively view the optimal value of each aggregated metric and other metric values.

数据查看

聚合指标 详细指标

搜索模型

模型名称	排名	AVG	低难度科学问...	高难度科学问...	跨学科综合能力	生物能力	哲学能力	科学问答综合...	多学科以及生...	生物和哲学能力	跨学科复杂综...	跨学科多综合...
hunyuan-lite-7b-256k	1	0.8656	0.8403	0.9539	0.8911	0.8619	0.7485	0.8971	0.8765	0.8052	0.8951	0.8868
hunyuan-lite-13b-32k	2	0.7971	0.7983	0.9945	0.8104	0.7092	0.6343	0.8964	0.7598	0.6718	0.8677	0.8281
hunyuan-lite-2b-32k	3	0.7190	0.5734	0.9996	0.7393	0.6876	0.5501	0.7865	0.7135	0.6189	0.7708	0.7500
DeepSeek-R1-Distill-Q...	4	0.7033	0.5639	0.9899	0.7290	0.6172	0.5922	0.7769	0.6731	0.6047	0.7609	0.7250
test-Token	5	0.5687	0.3073	0.7937	0.5827	0.6283	0.5101	0.5505	0.6055	0.5692	0.5612	0.5780
llama3.2_1b_chat	6	0.5538	0.2678	0.7641	0.5748	0.6269	0.5201	0.5160	0.6009	0.5735	0.5356	0.5584

You can quickly filter models in the search box in the upper-right corner to view the results of aggregated metrics.

数据查看

聚合指标 详细指标

hunyuan

模型名称	排名	AVG	低难度科学问...	高难度科学问...	跨学科综合能力	生物能力	哲学能力	科学问答综合...	多学科以及生...	生物和哲学能力	跨学科复杂综...	跨学科多综合...
hunyuan-lite-7b-256k	1	0.8656	0.8403	0.9539	0.8911	0.8619	0.7485	0.8971	0.8765	0.8052	0.8951	0.8868
hunyuan-lite-13b-32k	2	0.7971	0.7983	0.9945	0.8104	0.7092	0.6343	0.8964	0.7598	0.6718	0.8677	0.8281
hunyuan-lite-2b-32k	3	0.7190	0.5734	0.9996	0.7393	0.6876	0.5501	0.7865	0.7135	0.6189	0.7708	0.7500

©2013-2026 Tencent Cloud. All rights reserved.

Click the **Detailed Metrics** tab to view the detailed metric values of each model on the corresponding evaluation set.
It supports statistics on model participation in evaluations from the dimensions of aggregated metrics and evaluation sets, allowing you to intuitively check whether models have already been evaluated on key evaluation sets.

Note:

1. A comparison mode is selected for the radar chart by default, and it can be deselected in the upper-right corner. When it is selected, it indicates that the size is scaled from an absolute value to a relative value, making it easier to compare and display models.
2. Visualization display: You can click the model name on the right side not to display the data of the current model on the radar chart, and click it again to display its data. If you hover the mouse pointer over each point on the radar chart, the evaluation results of the current model on the corresponding data set will be displayed.
3. If you reselect an evaluation task, comparison data set, and comparison model, you can click **Apply**, and the information displayed on the chart on the right side will be updated based on your latest selections. Otherwise, every time you go to this page, the chart will display the corresponding information based on the configurations submitted to which you clicked Apply last time.

©2013-2026 Tencent Cloud. All rights reserved.

Evaluation Set Format Requirements

Last updated: 2026-01-15 13:29:07

Model Evaluation supports both automated evaluation and manual evaluation. Currently, automated evaluation offers two modes: Inference and Evaluation and Evaluation Only.

- Inference and Evaluation mode: upload an evaluation set (containing only questions and reference answers), and perform inference and complete judge model scoring in the Automated Evaluation functional module.
- Evaluation Only mode: upload an evaluation set with model inference results (questions, reference answers, and inference results), and use the judge model for scoring in the Automated Evaluation functional module.

The following sections introduce the evaluation set format requirements of the Inference and Evaluation and Evaluation Only modes.

Inference and Evaluation

Input Parameter Description

Parameter Name	Required	Parameter Description
messages	Required	Main content of an evaluation chat, including system (system settings), role (assistant or user), and content.
ref_answer	Optional	Reference answer of a model.
Other Fields	Optional	User-defined fields.

File Formats

Only the JSONL format is supported. Each piece of data should be a valid JSON format. For example:

1. **Multi-turn chat (excluding gt):**

```
{
    "messages": [
        {
            "role": "system",
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            "content": "Intelligent Assistant is an LLM developed by
xxx. xxx is a Chinese technology company that has been dedicated to
research related to LLMs."
        },
        {
            "role": "user",
            "content": "Hello!"
        },
        {
            "role": "assistant",
            "content": "Hello! Is there anything I can do for you"
        },
        {
            "role": "user",
            "content": "What is the sum of 1 and 1"
        }
    ],
    "ref_answer": "The answer is 2",
    "max_tokens": 4096,
    "extra_content": "a user-defined field. It can be referenced in a
judge model"
}
```

2. **Multi-turn chat (In the last sentence of messages, role is assistant, and the content will be automatically parsed as gt.):**

```
{
    "messages": [
        {
            "role": "system",
            "content": "ht is an LLM developed by xxx. xxx is a Chinese
technology company that has been dedicated to researches related to
LLMs."
        },
        {
            "role": "user",
            "content": "Hello!"
        },
        {
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            "role": "assistant",
            "content": "Hello! Is there anything I can do for you"
        },
        {
            "role": "user",
            "content": "What is the sum of 1 and 1"
        },
        {
            "role": "assistant",
            "content": "2"
        }
    ],
    "ref_answer": "The answer is 2",
    "extra_content": "a user-defined field. It can be referenced in a
judge model"
}
```

3. **Single-turn chat**

```
{
    "messages": [
        {
            "role": "user",
            "content": "What is the sum of 1 and 1"
        }
    ],
    "ref_answer": "The answer is 2",
    "extra_content": "a user-defined field. It can be referenced in a
judge model"
}
```

You can download evaluation set examples. For details, see Examples of Evaluation Sets in JSONL Format for Automated Evaluation.

In addition, the platform is compatible with the original formats for manual evaluation. Supported formats: JSONL and CSV. The evaluation set format is as follows:

- For evaluation sets in JSONL format: The format for each piece (row) of data is as follows:

©2013-2026 Tencent Cloud. All rights reserved.

```
{"system": "You are helpful.", "conversation": [{"prompt":
"712+165+223+711=","response": "1811"}]}
```

The system, prompt, and response fields correspond to the system input, prompt, and expected response respectively.
If a piece (row) of data does not have a system input, the following format can be used:

```
{"conversation": [{"prompt": "712+165+223+711=","response": "1811"}]}
```

- For evaluation sets in CSV format: An evaluation set consists of 3 columns with column names system, prompt, and response respectively. The system field is optional. If a piece (row) of data does not have a system input, the corresponding position should be left blank.
- You can download evaluation set examples. For details, see Examples of Evaluation Sets in JSONL Format for Manual Evaluation and Examples of Evaluation Sets in CSV Format for Manual Evaluation.

Evaluation Only

Input Parameter Description

Parameter Name	Required	Parameter Description
messages	Required	Main content of an evaluation chat, including system (system settings), role (assistant or user), and content.
model_outputs	Required	Inference result of a model.
ref_answer	Optional	Reference answer of a model.
Other Fields	Optional	User-defined fields.

File Formats

Only the JSONL format is supported. Each piece of data should be a valid JSON format. For example:

```
{
    "messages":
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    [
        {
            "role": "user",
            "content": "Hello!"
        },
        {
            "role": "assistant",
            "content": "Hello! Is there anything I can do for you"
        },
        {
            "role": "user",
            "content": "What is the sum of 16 and 16"
        }
    ],
    "ref_answer": "The answer is 32",
    "id": "141234314",
    "max_tokens": 1024,
    "temperature": 0.7,
    "top_p": 0.9,
    "top_k": 50,
    "model_outputs": // Above are the fields in the original data set.
    // This field is used to record the outputs of each model. The type is
a list, and each object in the list contains the following:
    // model_name: name of the model.
    // responses: output of the model.
    [
        {
            "model_name": "llama3",
            "responses":
            [
                {
                    "content": "The answer is 32."
                },
                {
                    "content": "32 is the answer."
                },
            ]
        },
        {
            "model_name": "qwen3",
            "responses":
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            [
                {
                    "content": "The answer is 32.",
                    "reasoning_content": "Wait..."
                },
                {
                    "content": "The answer is 32.",
                    "reasoning_content": "Thinking..."
                }
            ]
        }
    ]
}
```

Detailed Description of the model_outputs Field

This field is a list used to store inference results of different models. Each element in the list is an object, representing all outputs of a model. The object contains the following:

- model_name (string): name specified for the model to be tested.
- responses (list): contains a list of one or more inference results of the model.
- Each element in the list represents a complete inference output. The output contains the following:
  - content (string): final answer of the model.
  - reasoning_content (string, optional): thinking process before the model generates an answer.

Internal structure of the inference result object of the model (model_outputs):

Key Name	Type	Required	Description
model_name	String	Yes	Unique identifier of the model to be tested, for example, llama3-8b. Note: In the entire data set, the same name should be used for the same model so that the report can correctly summarize all the performance data of the model.

©2013-2026 Tencent Cloud. All rights reserved.

responses	Array	Yes	An array containing one or more inference results of the model. It supports scenarios where n is greater than 1. That is, inference is performed multiple times for the same input prompt.

Internal structure of the inference result object (responses):

Key Name	Type	Required	Description
content	String	Yes	Final result generated by the model to be tested. It is referenced during scoring based on a judge model.
reasoning_content	String	No	Used to store the thinking process, chain of thought, or any intermediate drafts before the model to be tested generates the final content. This content is provided to help with deeper analysis. If the model does not output this content, this key can be omitted.

Complete Example

```
// Line 1: contains the inference results of two models, where qwen2 has reasoning_content.
{"messages": [{"role": "user", "content": "What is the sum 16 and 16"}], "ref_answer": "The answer is 32", "id": "141234314", "model_outputs": [{"model_name": "llama3", "responses": [{"content": "The answer is 32."}]}, {"model_name": "qwen2", "responses": [{"content": "The answer is 32.", "reasoning_content": "The user is asking a simple math question. 16 + 16 equals 32."}]}]}
// Second line: contains two different inference results of a model (n=2).
{"messages": [{"role": "user", "content": "Write a poem praising programmers"}], "ref_answer": "Code is like poetry, expressing the beauty of logic and shaping a world using your fingertips.", "id": "141234315", "model_outputs": [{"model_name": "llama3", "responses": [{"content": "Ten lines of code can change the world."}, {"content": "Fingers dance and code likes a song."}]}]}
```

©2013-2026 Tencent Cloud. All rights reserved.

Automated Evaluation: Prompt for Judge Model Scoring and Preprocessing/Postprocessing Format Requirements

Last updated: 2026-01-15 13:28:14

Automated evaluation provides the feature of automatic scoring by a judge model, which significantly improves scoring efficiency and reduces the cost of manual scoring.
The **judge model scoring feature** provides a wizard-based and customizable way to define the evaluation process and configuration for scoring. **It allows users to directly customize the evaluation process based on the platform's built-in scoring nodes — preprocessing, judge model, and postprocessing.** It fully supports users in highly customizing scoring prompts and preprocessing/postprocessing scripts to achieve the best scoring results for the evaluation set.
The scoring nodes are defined as follows:

- **Judge model:** evaluates the evaluation set by using an AI model for scoring. Its core feature is to support users in choosing Tencent Cloud TI-ONE Platform (TI-ONE) online services or inputting API calls for scoring.
  - **Scoring prompt:** tailors appropriate scoring rules for the evaluation set through custom input methods. The rules primarily include the system definition of the judge model, scoring criteria, and the need to combine questions, reference answers, and outputs of the model under evaluation into a string, which is then input to the judge model. To achieve better scoring results, users are allowed to customize the scoring prompt for each judge model. The input content for the judge model needs to be dynamically constructed using Jinja2 template syntax.
- To obtain better evaluation results, in some cases, data needs to be processed before and after the judge model's scoring. The platform supports inputting Python scripts for data processing, which includes two types: preprocessing and postprocessing.
  - **Preprocessing:** typically used to process the raw data set and the output (response) of the model under evaluation, serving as the final scoring object input to the judge model.
  - **Postprocessing:** typically used to perform data processing on the output of the judge model to obtain the final scoring results.

The following sections detail the scoring prompt and preprocessing/postprocessing formats for the judge model.

I. Usage Guidelines for Scoring Template of the Judge Model

©2013-2026 Tencent Cloud. All rights reserved.

The scoring prompt is used to define the input for the judge model, including the system definition of the judge model, scoring criteria, and the need to combine questions, reference answers, and outputs of the model under evaluation into a string, which is then input to the judge model. Users need to dynamically construct the input content of the judge model using Jinja2 template syntax.

> **Note:**
> **What Is Jinja2?**
> Jinja2 is a powerful and flexible template engine that supports syntax such as variable interpolation, control statements, and filters, and is widely used for generating dynamic text content.
> In judge model evaluation, we use Jinja2 templates to dynamically construct the question and the response of the model under evaluation seen by the judge model, enabling it to make evaluations based on a clear and standardized context.

When editing the prompt, users can apply the corresponding field content from the response of the model under evaluation and the data of the evaluation set. Additionally, the platform by default uses the user data set and provides predefined variables for users to reference during editing.

1. Field Definitions and Variable Descriptions

Description of Predefined Variables

The platform predefines the following 3 core variables by default to quickly obtain key information in chats. Users can directly reference these variables in the template.

Variable Name	Definition Method	Data Source	Default Value
data.question	Content of the last message where the role is user	Extracts the content of the last message with role: "user" from a complete chat record (data.messages) in the data set.	None
data.gt	Content of the last message where the role is assistant	Extracts the content of the last message with role: "assistant" from a complete chat record (data.messages) in the data set.	Defaults to None if the role of the last message is not assistant.
data.history	Chat history (remaining messages	The text formed by sequentially concatenating the remaining messages	The default value is None for a single-turn chat.

©2013-2026 Tencent Cloud. All rights reserved.

	excluding question and gt)	after filtering out the last user and assistant messages from data.messages.	

Note:
All variables are derived from partial references to the evaluation set, as shown in the following example:

```
{
 "messages": [
  {"role": "user", "content": "What is the climate like in Beijing?"},
  {"role": "assistant", "content": "Beijing has a temperate monsoon climate..."}, // The above content constitutes data.history.
  {"role": "user", "content": "How about in summer?"},  // Last user message > data.question.
  {"role": "assistant", "content": "Summer is hot and rainy..."}  // Last assistant message > data.gt.
 ],
 "ref_answer": "Beijing's summer climate features high temperatures and abundant rainfall...",  // Optional field.
}
```

Description of Other Available Fields

Besides predefined variables, users can reference custom fields in the evaluation set based on task requirements (Note: ensure the field exists in the data of the evaluation set or the response of the model under evaluation). The following are commonly available fields:

Field Name	Definition and Purpose
data.ref_answer	Reference answer, corresponding to the ref_answer field in the original data (return None if not included in the original data).
response.content	Actual response content from the model under evaluation.
data.other_user_defined_fields	All user-defined data fields in the evaluation data can be referenced.
response.other_fields_returned_by_model	All other fields returned by the model under evaluation can be referenced; common examples include response.reasoning_content and response.tool_calls.

2. Example of Scoring Prompt Concatenation

©2013-2026 Tencent Cloud. All rights reserved.

Note:
When a judge model is used for evaluation, the input to the judge model must be a structured, clear string that includes system definitions, scoring criteria, and the combination of questions, reference answers, and the outputs from the model under evaluation. Since the evaluation set contains multiple entries, and the content of each question, reference answer, and output from the model under evaluation changes dynamically, the above content needs to be concatenated into the prompt for the judge model. Jinja allows users to define placeholders (variables) within a piece of text (the scoring prompt) and replace these variables with actual values by passing in a context dictionary, thereby dynamically generating the final scoring prompt to be input to the judge model.

Combining predefined variables and custom fields, the following provides template examples for several different types of tasks.

Example 1: Single-Turn Q&A Template Concatenation

- **Raw data sample**

```
"data": {
    "messages": [
      {"role": "user", "content": "What is the core content of Newton's
First Law?"},
      {"role": "assistant", "content": "Every object remains at rest, or
in motion at a constant speed in a straight line, unless it is acted
upon by a force"}
    ]
  }
```

- **Sample of response from the model under evaluation**

```
"response": {
    "content": "Newton's First Law states: An object will remain at rest
or in uniform motion in a straight line unless acted upon by an external
force.",
    "reasoning_content": "The user is asking about Newton's First Law.
First, I need to determine which discipline this belongs to..."
  }
```

- **Scoring prompt and concatenation result**

©2013-2026 Tencent Cloud. All rights reserved.

Scoring Prompt

Act as a fair judge to evaluate the quality of the AI assistant's response to the user's question shown below. You will be provided with the current user question, a reference answer, and the assistant's response.
When evaluating the assistant's response, you first need to compare it with the reference answer and identify any errors or inaccuracies in the assistant's response.
Then, consider whether the assistant's response is helpful, relevant, and concise.

- Helpful: The response correctly addresses the user's question or follows the request. If the user's question is ambiguous or could be interpreted in multiple ways, a more helpful and appropriate approach is to ask the user for clarification or more information, rather than answering based on assumptions.
- Relevant: Every part of the response is closely related to or appropriately corresponds to the question asked.
- Concise: The response is clear and straightforward, not verbose, wordy, or containing unnecessary information.

Then, if applicable, also consider the creativity and novelty in the assistant's response.
Finally, identify any important information missing from the assistant's response that would help respond to the user's question, especially content present in the reference answer.
Your evaluation should comprehensively consider the following factors: compared with the reference answer, assess the assistant's response in terms of correctness, helpfulness, relevance, accuracy, depth, creativity, and level of detail.
First, provide a detailed evaluation note, focusing on analyzing how well the assistant's response matches the reference answer, or whether it is superior to the reference answer in certain aspects. Try to be as objective as possible.
After completing the note, strictly follow the format below to rate the assistant's response on a scale of 1 to 10:
"Score: [[score]]", for example: "Score: [[5]]".

[User question]
{{ data.question }}

[Reference answer]
{{ data.gt }}

[Assistant response]
{{ response.content }}

[Model inference process]
{{ response.reasoning_content }}

©2013-2026 Tencent Cloud. All rights reserved.

Concatenation Result

Act as a fair judge to evaluate the quality of the AI assistant's response to the user's question shown below. You will be provided with the current user question, a reference answer, and the assistant's response and reasoning process.
When evaluating the assistant's response, you first need to compare it with the reference answer and identify any errors or inaccuracies in the assistant's response.
Then, consider whether the assistant's response is helpful, relevant, and concise.

- Helpful: The response correctly addresses the user's question or follows the request. If the user's question is ambiguous or could be interpreted in multiple ways, a more helpful and appropriate approach is to ask the user for clarification or more information, rather than answering based on assumptions.
- Relevant: Every part of the response is closely related to or appropriately corresponds to the question asked.
- Concise: The response is clear and straightforward, not verbose, wordy, or containing unnecessary information.

Then, if applicable, also consider the creativity and novelty in the assistant's response.
Finally, identify any important information missing from the assistant's response that would help respond to the user's question, especially content present in the reference answer.
Your evaluation should comprehensively consider the following factors: compared with the reference answer, assess the assistant's response in terms of correctness, helpfulness, relevance, accuracy, depth, creativity, and level of detail.
First, provide a detailed evaluation note, focusing on analyzing how well the assistant's response matches the reference answer, or whether it is superior to the reference answer in certain aspects. Try to be as objective as possible.
After completing the note, strictly follow the format below to rate the assistant's response on a scale of 1 to 10:
"Score: [[score]]", for example: "Score: [[5]]".

[User question]
What is the core content of Newton's First Law?

[Reference answer]
Every object remains at rest, or in motion at a constant speed in a straight line, unless it is acted upon by a force.

[Assistant response]

©2013-2026 Tencent Cloud. All rights reserved.

Newton's First Law states: An object will remain at rest or in uniform motion in a straight line unless acted upon by an external force.

[Model inference process]
The user is asking about Newton's First Law. First, I need to determine which discipline this belongs to...

Example 2: Multi-Turn Q&A Template Concatenation

- **Raw data sample**

```
 "data": {
    "messages": [
      {"role": "system", "content": "You are an intelligent assistant
developed by Company A, capable of accurately answering user
questions."},
      {"role": "user", "content": "Hello, could you help explain the
general theory of relativity to me?"},
      {"role": "assistant", "content": "The general theory of relativity
was proposed by Einstein, introducing the relationship between the
gravitational field and the curvature of spacetime."},
      {"role": "user", "content": "Then what is special relativity?"},
      {"role": "assistant", "content": "When an object moves at speeds
approaching the speed of light, time, space, mass, and energy all
exhibit laws entirely different from everyday experience, and the speed
of light is the ultimate speed limit for all objects in the universe."}
    ]
  }
```

- **Sample of response from the model under evaluation**

```
"response": {
    "content": "Special relativity primarily studies the laws of motion
in inertial frames, including time dilation and length contraction.",
    "reasoning_content": "The user is asking about special relativity,
so the core content should be explained. Common expressions include the
nature of spacetime under high-speed motion, such as time dilation and
length contraction."
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    }
```

- **Scoring prompt and concatenation result**

Scoring Prompt

Please act as a fair judge to evaluate the quality of the AI assistant's response to the user's question shown below. You will be provided with the chat history between the assistant and the user, the current user question, a reference answer, and the assistant's response.
When evaluating the assistant's response, you first need to compare it with the reference answer and identify any errors or inaccuracies in the assistant's response. Additionally, check whether the assistant's response aligns with the information provided in the chat history, as well as how closely it matches the reference answer.
Then, evaluate whether the assistant's response is helpful, relevant, and concise.

- Helpful: The response correctly addresses the user's current question or follows the request, while providing appropriate replies based on the chat context (if applicable). If the user's question is ambiguous or could be interpreted in multiple ways, a more helpful and appropriate approach is to ask the user for clarification or more information, rather than answering based on assumptions.
- Relevant: Every part of the response is closely related to or appropriately addresses the question asked, considering both the current question and the relevant contextual information from the chat history.
- Concise: The response is clear and straightforward, not verbose, wordy, or containing unnecessary information.

Then, if applicable, also consider the creativity and novelty in the assistant's response.
Finally, identify any important information missing from the assistant's response that would help respond to the user's question, especially content present in the reference answer or information from the chat history that should have been referenced but was not mentioned.
Your evaluation should comprehensively consider the following factors: compared with the reference answer, assess the assistant's response in terms of correctness, helpfulness, relevance, accuracy, depth, creativity, and level of detail.
First, provide a detailed evaluation note, focusing on analyzing how well the assistant's response matches the reference answer, or whether it is superior to the reference answer in certain aspects. Try to be as objective as possible.
After completing the note, strictly follow the format below to rate the assistant's response on a scale of 1 to 10:
"Score: [[score]]", for example: "Score: [[5]]" or "Score: [[8]]".

[Chat history]
{{ data.history }}

©2013-2026 Tencent Cloud. All rights reserved.

[User question]
{{ data.question }}

[Reference answer]
{{ data.gt }}

[Assistant response]
{{ response.content }}

Concatenation Result

Please act as a fair judge to evaluate the quality of the AI assistant's response to the user's question shown below. You will be provided with the chat history between the assistant and the user, the current user question, a reference answer, and the assistant's response.
When evaluating the assistant's response, you first need to compare it with the reference answer and identify any errors or inaccuracies in the assistant's response. Additionally, check whether the assistant's response aligns with the information provided in the chat history, as well as how closely it matches the reference answer.
Then, evaluate whether the assistant's response is helpful, relevant, and concise.

- Helpful: The response correctly addresses the user's current question or follows the request, while providing appropriate replies based on the chat context (if applicable). If the user's question is ambiguous or could be interpreted in multiple ways, a more helpful and appropriate approach is to ask the user for clarification or more information, rather than answering based on assumptions.
- Relevant: Every part of the response is closely related to or appropriately addresses the question asked, considering both the current question and the relevant contextual information from the chat history.
- Concise: The response is clear and straightforward, not verbose, wordy, or containing unnecessary information.

Then, if applicable, also consider the creativity and novelty in the assistant's response.
Finally, identify any important information missing from the assistant's response that would help respond to the user's question, especially content present in the reference answer or information from the chat history that should have been referenced but was not mentioned.
Your evaluation should comprehensively consider the following factors: compared with the reference answer, assess the assistant's response in terms of correctness, helpfulness, relevance, accuracy, depth, creativity, and level of detail.

©2013-2026 Tencent Cloud. All rights reserved.

First, provide a detailed evaluation note, focusing on analyzing how well the assistant's response matches the reference answer, or whether it is superior to the reference answer in certain aspects. Try to be as objective as possible.
After completing the note, strictly follow the format below to rate the assistant's response on a scale of 1 to 10:
"Score: [[score]]", for example: "Score: [[5]]" or "Score: [[8]]".

[Chat history]
[SYSTEM] You are an intelligent assistant developed by Company A, capable of accurately answering user questions.
[USER] Hello, could you help explain the general theory of relativity to me?
[BOT] The general theory of relativity was proposed by Einstein, introducing the relationship between the gravitational field and the curvature of spacetime.

[User question]
Then what is special relativity?

[Reference answer]
When an object moves at speeds approaching the speed of light, time, space, mass, and energy all exhibit laws entirely different from everyday experience, and the speed of light is the ultimate speed limit for all objects in the universe.

[Assistant response]
Special relativity primarily studies the laws of motion in inertial frames, including time dilation and length contraction.

II. Specifications for Preprocessing/Postprocessing Script Writing

When you need to customize the preprocessing logic for evaluation data or the postprocessing logic for the model output response, you must provide a Python script that complies with the following specifications. The evaluation system will import and execute the agreed-upon functions in your script as a module.

1. Function Naming and API Conventions

Preprocessing and postprocessing scripts must each contain the following two functions: preprocess and postprocess. The evaluation module will call the preprocessing functions based on these names.

2. Preprocessing Function

©2013-2026 Tencent Cloud. All rights reserved.

The preprocessing function is typically used to process the raw data set and the response of the model under evaluation to construct the final prompt input to the model. You need to implement this function.

- Function name: preprocess
- Feature: Process the response of the model under evaluation or the raw data set, ultimately returning a value for use by subsequent evaluation modules or returning no value.
- Input parameters: data, resp, **kwargs
- Output parameter: processing result, which serves as the result of the current processing node [displayed in the processing results].
- Content of the data structure: Users can add new fields to the structure for subsequent reference via the dictionary.
  - data [dict type]: Automatically parsed based on the content of the evaluation set, containing the messages field in OpenAI's chat format. When data is prepared according to the evaluation format, any other fields will be included in this structure, such as the ref_answer field.

Field Name	Included or Not	Field Description	Field Example
messages	Yes	OpenAI chat format	"messages": [{"role": "user", "content": "Hello"}, {"role": "assistant", "content": "Hello, how can I help you"}, {"role": "user", "content": "What is the sum of 12 and 12"}, {"role": "assistant", "content": "24"}]
ref_answer	Parsed from the user data set; None if the original data set does not contain this field.	Reference answer	"The answer is 24"
gt	If the last message in the messages field has the role of assistant, then its content is taken as this field.	Standard answer	"24"
question	Parse the content of the last message where the role is user in the messages field as the content of this field.	User question	"What is the sum of 12 and 12"

©2013-2026 Tencent Cloud. All rights reserved.

history	The remaining content after eliminating the gt and question fields from messages, rendered as a string according to the Jinja template of the platform.	Chat history	"[USER] Hello [BOT] Hello, how can I help you"
Other Fields	Other fields in user data. Users can also add them to processing scripts.	Custom fields	str | int | float | dict and other JSON-serializable content

- response [dict type]: The parsed result of the LLM's response, containing the content field. If the inference framework supports separate_reasoning, it includes the reasoning_content field. Other possible fields include tool_calls.

Field Name	Included or Not	Field Description	Field Example
content	Yes	Inference result of the LLM	"The answer is 24"
reasoning_content	Included if the inference framework supports separate display and return of the chain of thought (CoT); otherwise, the value is None.	Content of LLM inference CoT	"Hmm, the user is asking 'What is the sum of 12 and 12'. This seems like a basic question, but someone just starting to learn math might need a detailed explanation..."
tool_calls	Included if tool calls are executed and this field is returned; otherwise, the value is None.	Content of the LLM tool call	[{"id":"call_id","type":"function","function": {"name":"get_current_weather","arguments":" {\"location\": \"San Francisco, USA\", \"format\": \"celsius\"}"}}]
Other Fields	Other fields returned by the model. Users can also add them to processing scripts.	Custom fields	str | int | float | dict and other JSON-serializable content.

- Function signature:

©2013-2026 Tencent Cloud. All rights reserved.

```
def preprocess(data: dict, resp: dict, **kwargs) -> bool | int | str |
float | None:
    """
   :data: Dict, a dictionary containing the following key information:
                    - 'messages': Multi-turn chat history.
                    - 'ref_answer': (Optional) Reference answer.
                    - And all other user-defined fields
    :resp: Dict, the inference result of the LLM, containing the
following:
                    - content: Model-generated response.
                    - reasoning_content: Inference process of the model-
generated response (if any).
                    - tool_calls: List of tools called by the model (if
any)
                    - And all other user-defined fields
    :**kwargs: Extension fields of the platform, which may not be
provided during preprocessing.
    :return: Preprocessing result, which can be referenced in the jinja
template for judge model evaluation.
    """
    # --- Implement your logic here. ---
    pass
```

Note:

1. The preprocessing function must include the fields data, resp, and **kwargs. The **kwargs field is used for future platform extensions to provide more operable data items.
2. data and resp support accessing or adding fields via dictionary/object methods.
3. To prevent the evaluation process from being interrupted due to unexpected data formats, it is recommended to incorporate basic exception handling in the script.

```
def preprocess(data: dict, resp: dict) -> bool | int | str |
float | None:
    try:
        # Your processing logic
    except Exception as e:
        print(f"[Error in the preprocessing script]: {e}")
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        return 1  # Return the default value as needed.
```

- Usage examples:
  - Example 1: Determine whether the model's response contains a CoT.

```
def preprocess(data: dict, resp: dict) -> bool:
    # Check whether the model response contains reasoning_content.
    return resp['reasoning_content'] == ''
```

  - Example 2: Remove the CoT from the content.

```
import re
default_cot_pairs = [('<think>', '</think>'), ('<Think>',
'</Think>')]
def remove_cot(text: str, cot_pairs=None) -> str:
    if cot_pairs is None:
        cot_pairs = default_cot_pairs
    ret = text
    for t0, t1 in cot_pairs:
        ret = re.sub(f"{t0}.*?{t1}", '', ret, flags=re.DOTALL)
    return ret.strip()


def preprocess(data: dict, resp: dict) -> bool | int | str | float |
None:
    field = 'clean'
    resp[field] = remove_cot(resp['content'])
    return resp[field]
```

  - Perform data processing only, without returning a value.

```
def preprocess(data: dict, resp: dict) -> None:
    # Example: Add the content of the model response to the data
    if resp['content']:
        data['resp'] = resp['content']
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    # Does not affect the evaluation judgment, and returns None.
    return None
```

3. Postprocessing Function

The postprocessing function is typically used to perform postprocessing on the output of the judge model to obtain the final scoring results.

- Function name: preprocess
- Feature: Process the response of the judge model.
- Input parameters [required]: judge_reqs, judge_resps, judge_models, data, resp, **kwargs
- Output parameter: processing result, which serves as the result of the current processing node [displayed in the processing results].
- Content of the data structure: Users can add new fields to the structure for subsequent reference via the dictionary.

  judge_reqs [list[dict] type]: A list where inference requests for judge models are added sequentially according to the processing order defined in the evaluation metric configuration. Each item in the list is generated by parsing the user's prompt template and configuration of inference hyperparameters, and by default contains the messages field with stream enabled, and the separate_reasoning field.

Field Name	Included or Not	Field Description	Field Example
messages	Yes	OpenAI chat format.	[{"role": "system", "content": "You are a helpful assistant. 0827-adjustment" },{"role": "user","content": "You are a judge. Please score the response.\n\n\n[Historical information and question]\n\n<|im_start|>user\nHello<|im_end|>\n\n\n<|im_start|>assistant\nHello, how can I help you?<|im_end|>\n\n\n<|im_start|>user\nWhat is the sum of 18 and 18<|im_end|>\n\n\n\n[Answer from the to-be-evaluated model]\n{'content': '18 + 18 = 36'}\n\n[Reference answer]\nThe answer is 24\n\n\nProvide your score, with a maximum of 5 and a minimum of 1."}]
Other Fields	Other fields returned by the model. Users can also add them to processing scripts.	Custom fields	str | int | float | dict and other JSON-serializable content

  - judge_models [list[dict] type]: A list where each item contains the basic configuration of a judge model, added sequentially according to the processing order defined in the evaluation metric configuration.

Field Name	Included or Not	Field Description	Field Example
name	Yes	Name of the judge model	"deepseek"
judge_template_content	Yes	Scoring template for the judge model	"""Scoring template for the judge, for example: You are a judge. Please score the response.

©2013-2026 Tencent Cloud. All rights reserved.

			[Historical information and question] {% for message in data.messages %} {{'<|im_start|>' + message['role'] + '\n' + message['content'] + '<|im_end|>' + '\n'}} {% endfor %} [Answer from the to-be-evaluated model] {{ response.content }} [Reference answer] {{ data.ref_answer }} Provide your score, with a maximum of {{ max_score }} and a minimum of {{ min_score }}. """
generation_params	Not necessarily, parsed based on user configuration.	Inference parameters of the judge model	{\"temperature\": 0.8, \"top_p\": 0.85}
system_prompt	Not necessarily, parsed based on user configuration.	System prompt for judge model inference	"You are a helpful assistant. "

  - data [dict type]: The data structure is the same as the data structure in preprocessing.
  - judge_resps [list[dict] type]: A list where the inference results of the judge models are added sequentially according to the processing order defined in the evaluation metric configuration. The structure of each item in the list is the same as the resp structure in preprocessing.
- Function signature:

```
def postprocess(
        judge_reqs: list[dict],
        judge_resps: list[dict],
        judge_models: list[dict],
        data: dict,
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        resp: dict,
        **kwargs,
) -> bool | int | str | float | None:
    """
    :param judge_reqs: Scoring request data for the judge model.
    :param judge_resps: Inference results of the judge model.
    :param judge_models: Relevant configuration for the judge model.
    :return: performs custom postprocessing logic and returns a value of
type bool | int | str | float | None for final evaluation result
judgment, display, or statistics to facilitate metric calculation.
    """

    # --- Implement your logic here. ---
    pass
```

Note:

The postprocessing function must include the fields judge_reqs, judge_resps, judge_models, data, resp, and **kwargs. The **kwargs parameter is used for future platform extensions to provide more operable data items. By default, the judge_req, judge_resp, and judge_model of the last node of the judge model are injected and can be accessed as follows:

```
def postprocess(
        judge_reqs: list[dict],
        judge_resps: list[dict],
        judge_models: list[dict],
        data: dict,
        resp: dict,
        **kwargs,
) -> bool | int | str | float | None:
    judge_req = kwargs.get('judge_req')
    judge_resp = kwargs.get('judge_resp')
    judge_model = kwargs.get('judge_model')
```

- Usage examples:
  - Calculate the average score from all judges.

©2013-2026 Tencent Cloud. All rights reserved.

```
def postprocess(judge_reqs, judge_resps, judge_models, data, resp,
**kwargs):
    scores = []
    for jr in judge_resps:
        try:
            # Assume the judge returns "The answer scores 4 points"
or "Score: 4".
            content = jr['content']
            score = int(content.strip().split(" ")[-2])  # Simple
example; adjust according to the actual format.
            scores.append(score)
        except:
            continue
    if scores:
        return round(sum(scores) / len(scores), 2)
    return None
```

- Determine whether the majority of judges consider the answer correct.

```
def postprocess(judge_reqs, judge_resps, judge_models, data, resp,
**kwargs):
    positive_count = 0
    for jr in judge_resps:
        content = jr['content']
        if "good" in content or "correct" in content or "4" in
content or "5" in content:
            positive_count += 1
    return positive_count > len(judge_resps) / 2  # Return true if
more than half consider it correct.
```

- Perform data organization only, and return None.

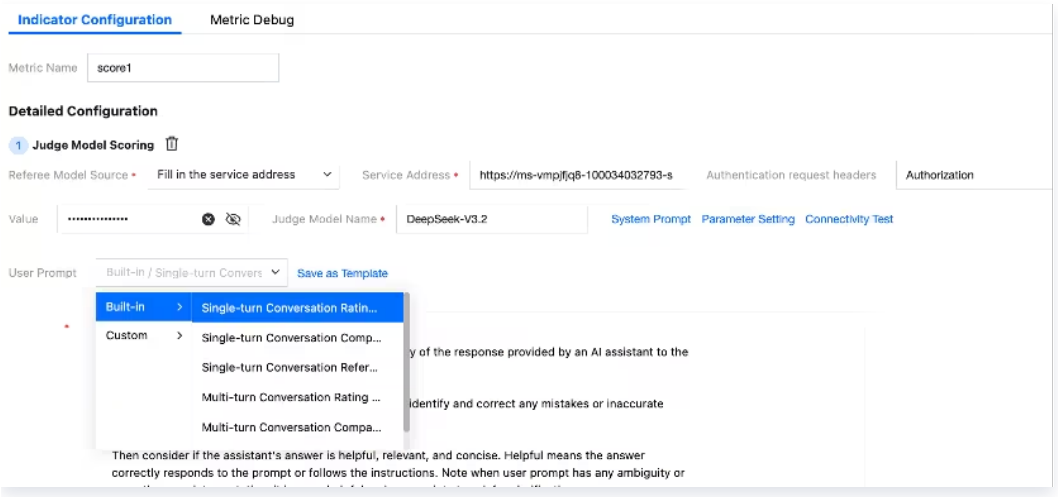
```
def postprocess(judge_reqs, judge_resps, judge_models, data, resp,
**kwargs):
    # Data storage, structure transformation, and so on, can be
performed here without affecting the final evaluation.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    return None
```

III. Platform Built-in Scoring Prompt Template and Recommended Postprocessing Scripts

Overview: To improve user configuration efficiency, the platform has built-in scoring prompt templates for single-turn and multi-turn chats, along with corresponding postprocessing methods.



1. Scoring Method for Single-Turn Chats

This method directly scores the responses of the model under evaluation, with a final score ranging from 1 to 10. If no custom postprocessing script is defined, the platform will help users perform postprocessing by default to extract the specific score as the final result.

2. Comparative Scoring Method for Single-Turn Chats

Based on the original question, the judge model scores by comparing the reference answer (the ref_answer field in the data set; users may modify this to other custom fields) with the response from the model under evaluation. The judge model outputs whether A or B is better, worse, or equal. The postprocessing script then extracts this result and maps it to the corresponding score.
The corresponding postprocessing script is provided below for user reference:

```
import re
from typing import Optional
ans_map = {
    "A>>B": 1,
    "A>B": 2,
    "A=B": 3,
    "B>A": 4,
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    "B>>A": 5,
}
def parse_last_comparison(text: str) -> Optional[str]:
    pattern = r"\[\[(A>>B|A>B|A=B|B>A|B>>A)\]\]"
    matches = re.findall(pattern, text)
    return matches[-1] if matches else None
def postprocess(
        judge_reqs: list[dict],
        judge_resps: list[dict],
        judge_models: list[dict],
        data: dict,
        resp: dict,
        **kwargs,
) -> Optional[int]:
    last_comparison = parse_last_comparison(judge_resps[-1]['content'])
    return ans_map.get(last_comparison)
```

3. Reference-based Scoring Method for Single-Turn Chats

This method scores the responses of the model under evaluation by referring to the ref_answer field, with a final score ranging from 1 to 10. If no custom postprocessing script is defined, the platform will help users perform postprocessing by default to extract the specific score as the final result.

4. Scoring Method for Multi-Turn Chats

This method scores the responses of the model under evaluation (applicable to usage scenarios requiring context), with a final score ranging from 1 to 10. If no custom postprocessing script is defined, the platform will help users perform postprocessing by default to extract the specific score as the final result.

5. Comparative Scoring Method for Multi-Turn Chats

This method scores the responses of the model under evaluation (applicable to usage scenarios requiring context). The judge model outputs whether A or B is better, worse, or equal. The postprocessing script then extracts this result and maps it to the corresponding score.
The corresponding postprocessing script is provided below for user reference:

```
import re
from typing import Optional
ans_map = {
    "A>>B": 1,
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    "A>B": 2,
    "A=B": 3,
    "B>A": 4,
    "B>>A": 5,
}
def parse_last_comparison(text: str) -> Optional[str]:
    pattern = r"\[\[(A>>B|A>B|A=B|B>A|B>>A)\]\]"
    matches = re.findall(pattern, text)
    return matches[-1] if matches else None
def postprocess(
        judge_reqs: list[dict],
        judge_resps: list[dict],
        judge_models: list[dict],
        data: dict,
        resp: dict,
        **kwargs,
) -> Optional[int]:
    last_comparison = parse_last_comparison(judge_resps[-1]['content'])
    return ans_map.get(last_comparison)
```

6. Reference-based Scoring Method for Multi-Turn Chats

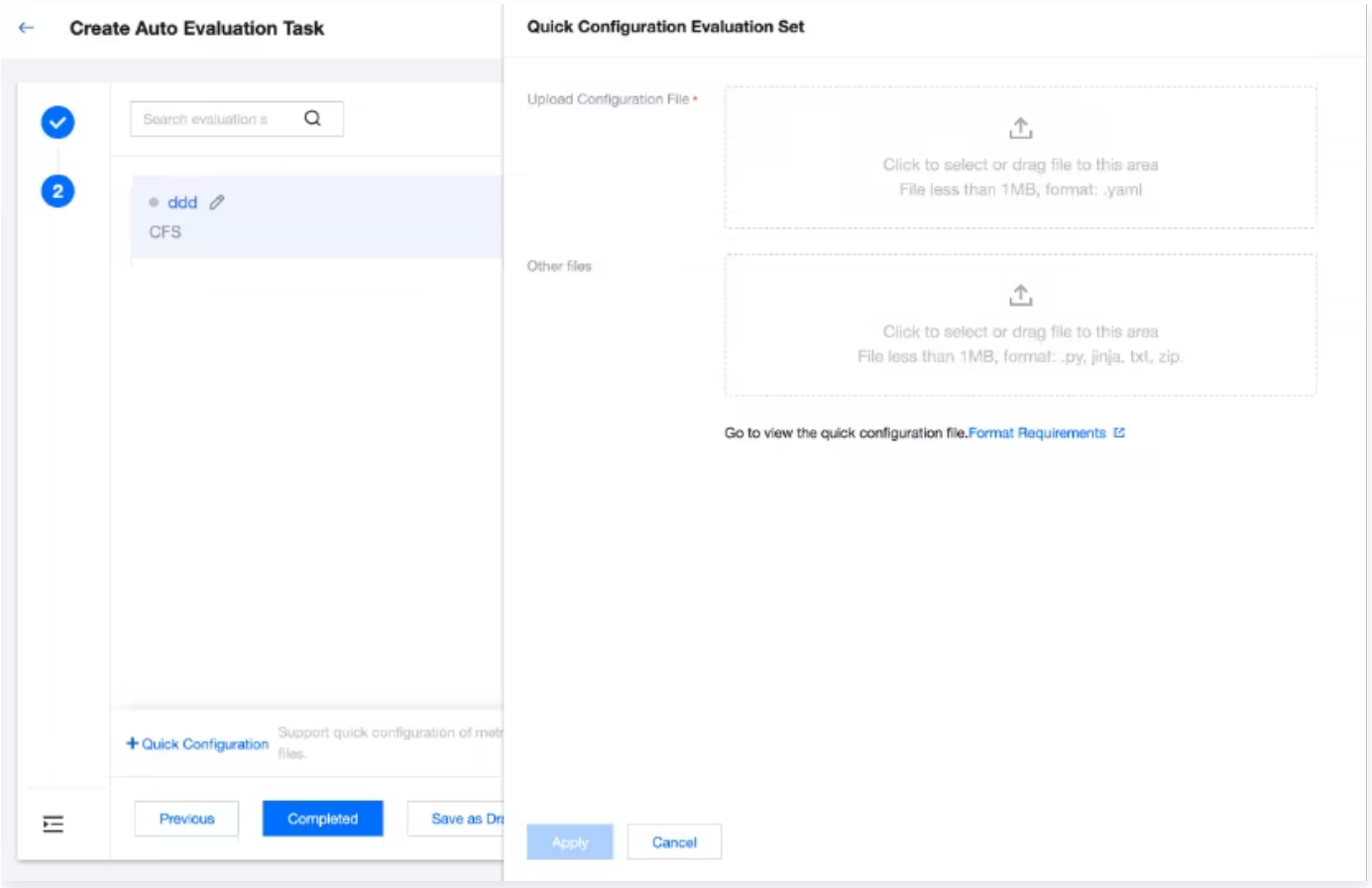
This method scores the responses of the model under evaluation by referring to the ref_answer field (applicable to usage scenarios requiring context), with a final score ranging from 1 to 10. If no custom postprocessing script is defined, the platform will help users perform postprocessing by default to extract the specific score as the final result.

©2013-2026 Tencent Cloud. All rights reserved.

Automated Evaluation: Quick Configuration File Specifications for Metrics

Last updated: 2026-01-15 13:25:25

You can **quickly configure metrics by uploading a file** on the automated evaluation configuration page. The file must include the evaluation set, the corresponding metric names for the evaluation set, and the detailed configuration information corresponding to each metric name (such as judge model information, scoring prompts, and preprocessing and post-processing scripts). You can click **Quick Configuration** to upload a custom YAML configuration file and any files that need to be referenced. Once the upload is completed, click **Apply**. The platform will automatically populate the configuration fields on the page by matching the evaluation set name that you specify with the metrics and their configuration information defined in the YAML configuration file.



Requirements for the YAML Configuration File Structure

The YAML file must clearly define the evaluation rules for the evaluation set. Examples of core fields are as follows:

©2013-2026 Tencent Cloud. All rights reserved.

```
- data_name: your_dataset_name  # Data set name. This field is associated
with the name entered on the evaluation set configuration page.
  metrics:  # Configuration for one or more metrics included in this data
set.
      # name: metric name.
    - name: judge model scoring.
      # The pipeline field is used to define the scoring process, where
each element in this array corresponds to a processing node in sequence.
      pipeline:
        # The type field is set to PREPROCESS, which indicates that this
node is a preprocessing node.
        - type: PREPROCESS
          # filename specifies the name of the Python script on which
preprocessing or postprocessing depends.
          # This field should be filled with the relative path of the
corresponding file to the root directory within the compressed package.
          filename: scripts/preprocessor.py
          # file_content: For preprocessing or postprocessing steps, you
can directly enter the script content in this field.
          file_content: |-
            def preprocess(data, resp, **kwargs) -> bool | int | str |
float | None:
                pass

        # type is JUDGE_MODEL, which indicates that this node is a scoring
node based on a judge model.
        - type: JUDGE_MODEL
          # Configure the judge model information by specifying the
judge_model field.
          judge_model:
            name: DeepSeek-R1  # Name of a judge model.
            # Method for calling a judge model:
            #   If MS is entered, it indicates using the Online Services
module of Tencent Cloud TI-ONE Platform (TI-ONE).
            #   If a URL is entered, it indicates using a third-party URL.
            source: MS
            # ti_model_service_api: This field is required if the Online
Services module of TI-ONE is used.
            ti_model_service_api:
```

©2013-2026 Tencent Cloud. All rights reserved.

```
              # service_group_id: ID of an online service, for example,
ms-45mrs4rv.
              service_group_id: ms-xxxx
              # service_group_name: Name of an online service.
              service_group_name: "DeepSeek-R1 judge model"
              # service_id: ID of an online service. You can choose Online
Service Details > Service Management to view the ID of a version, for
example, ms-45mrs4rv-1.
              # If this field is left unspecified, service_group_id with
the suffix -1 will be generated by default.
              service_id: ms-xxxx-1
              # url_prefix: address of Regular Service Calling. You can
choose Online Service Details > Service Call > Regular Service Calling to
view it.
              url_prefix: http://ms-xxxx-uuuu-sw.gw.ap-
region.ti.tencentcs.com/ms-xxxx
              # path_suffix: path for the calling chat API exposed by an
online service.
              path_suffix: /v1/chat/completions
              # auth_token: authentication token. You can choose Service
Authentication > Online Service Details to view it.
              auth_token: "aaaaaaaa"
            # third_party_api: If URL is specified for calling, this field
is required.
            third_party_api:
              # url: calling URL.
              url: http://ms-xxxx-uuuu-sw.gw.ap-
region.ti.tencentcs.com/ms-xxxx/v1/chat/completions
              # authorization_header: authentication HTTP header.
              authorization_header:
                # key HTTP Header Key
                key: authorization
                # value: HTTP header content.
                value: your_token
            # generation_params: parameters for calling a judge model.
            # Note: According to restrictions in Tencent Cloud API
specifications, this field should be a string.
            # Namely, add `|-` at the end of this field to limit the field
type to a string.
            # During subsequent parsing, we will convert this field into
one of the parameters in the judge model request body.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            generation_params: |-
              temperature: 0.8
              top_p: 0.85

            # judge_template_filename: scoring template file of a judge
model.
            judge_template_filename: template.jinja
            # judge_template_content: If no file is specified, you can use
this field to specify the scoring template content of a judge model.
            judge_template_content: |-
              You are a judge. Please rate the answers, with 5 as the
highest score and 1 as the lowest score.

              [Question]
              {{ data.question }}

              [Answer from the to-be-evaluated model]
              {{ response.content }}

              [Reference answer]
              {{ data.ref_answer }}

              Now give your score.

        # type is POSTPROCESS, which indicates that this node is a
postprocessing node.
        - type: POSTPROCESS
          filename: scripts/postprocessor.py
- data_name: another_dataset_name    # You can configure multiple data
sets in a YAML configuration file.
  metrics:
    - name: judge model scoring.
      pipeline:
        - type: PREPROCESS
          filename: preprocessor-at-root.py
        - type: JUDGE_MODEL
          filename: judge.yaml  # You can also configure a judge model
separately through a YAML file.
        - type: POSTPROCESS
          filename: postprocessor-at-root.py
```

©2013-2026 Tencent Cloud. All rights reserved.

File Upload Specifications

Other files referenced in the YAML configuration file can be uploaded through the frontend. You can select files one by one to upload or compress the files into a ZIP package to upload.
File download examples:

- Example of uploading configuration files in batches of a judge model
- Example of uploading dependency files in batches of a judge model

©2013-2026 Tencent Cloud. All rights reserved.

Introduction to Built-in Open-Source Evaluation Sets

Last updated: 2026-01-15 13:23:17

The platform offers a wide range of built-in open-source evaluation sets. The following introduces some built-in open-source evaluation sets and the actual scenarios of evaluation models, which provides reference for you during evaluation:

Mathematics

1. **GSM8K**
   - **Introduction:**

     GSM8K is a data set that contains approximately 8,500 grade-school math problems. It is primarily designed to test basic mathematical inference capabilities. The problems involve elementary arithmetic knowledge such as addition, subtraction, multiplication, division, fractions, and decimals.
   - **Application:**

     GSM8K is mainly used to evaluate the capabilities of models to solve grade-school math problems, especially the multi-step mathematical inference capabilities. For example, a model must understand the textual description in a math problem, convert it into mathematical expressions, and output a final answer through multi-step calculations.
   - **Actual scenarios:**

     In the education field, GSM8K can be used to develop intelligent tutoring systems that help students practice basic math problems and provide detailed solution steps.
     In the AI research field, GSM8K is an important benchmark for testing whether a model has basic logical inference capabilities.
2. **MATH**
   - **Introduction:**

     MATH is a data set that covers math problems with different difficulty ratings from elementary school to senior high school. It contains 12,500 math problems and can be used to test the advanced mathematical capabilities of models.
   - **Application:**

     MATH is used to evaluate the capabilities of models in solving complex mathematical problems, covering advanced mathematical knowledge such as algebra, geometry, and calculus.
   - **Actual scenarios:**

     In academic research, MATH can be used to test whether a model can handle math problems in high school or college, such as solving equations or proving geometric

©2013-2026 Tencent Cloud. All rights reserved.

theorems.
In the field of educational technology, MATH can be used to develop advanced mathematics learning tools that help students understand complex mathematical concepts.

3. **MATH-500**
   - **Introduction:**

     MATH-500 is a curated subset (sourced from international mathematics competitions such as American Mathematics Competitions (AMC) or American Invitational Mathematics Examination (AIME)) of the classic mathematical inference data set (MATH). It contains 500 competition-level math problems spanning topics such as algebra, geometry, number theory, and combinatorics from high school to early college levels. The math problems emphasize multi-step logical inference and creative problem-solving policies, and require the application of theorems, constructive proofs, and non-standard solution techniques.
     Its features include:
     MATH-500 provides complete problem-solving steps in LaTeX format for each math problem. Difficulty levels range from 1 to 5, with 5 being the most difficult. The highest accuracy achieved by human experts is only 90%, namely, the level of International Mathematical Olympiad (IMO) winners who have won gold medals.
   - **Application:**

     MATH-500 is mainly used to evaluate the extreme capabilities of LLMs in complex mathematical inference. It specifically tests: symbolic operations (such as polynomial simplification and inequality proofs), abstract modeling (translating verbal descriptions into mathematical structures),
     and long-chain inference (maintaining logical consistency beyond 15 steps).
   - **Actual scenarios:**

     MATH-500 provides adaptive generation of question banks and solution path feedback for competition preparation platforms such as the Art of Problem Solving (AoPS) community.
4. **AIME2024**
   - **Introduction:**

     The American Invitational Mathematics Examination (AIME) is a high-difficulty invitational competition for American Mathematics Competitions (AMC) contest winners. AIME 2024 or 2025 is widely used as a gold-standard test set for evaluating the mathematical inference capabilities of LLMs. Its characteristics are as follows: AIME focuses on core Olympiad areas such as number theory, combinatorial optimization, and geometric proofs; AIME emphasizes intuitive insight and calculation efficiency (answer questions within a limited time period); AIME features questions that demand filling in the blanks with integers to avoid linguistic expression interference.
   - **Application:**

©2013-2026 Tencent Cloud. All rights reserved.

AIME scores have become a key metric for measuring whether an LLM has achieved a fundamental leap in inference capabilities. For example:
DeepSeek-R1-0528 (updated on May 2025): AIME 2025 accuracy jumped from 70% to 87.5%, and AIME achieved deep thinking by extending the chain of thought to 23,000 tokens per problem.

- **Actual scenarios:**

  Algorithm optimization sandbox: drives innovation in the Reinforcement Learning from Human Feedback (RLHF) technology. For example, entropy regularization methods such as Clip-Cov and KL-Cov effectively prevent entropy collapse in reinforcement learning.

5. **TheoremQA**
   - **Introduction:**

     TheoremQA is a theorem-driven question answering benchmark. It is an evaluation data set focused on mathematical theorem inference, covering more than 350 high-difficulty problems in areas such as geometry, number theory, and calculus. Its unique value lies in requiring models to perform inference strictly using mathematical theorems, formulas, and proof logic, rather than relying on statistical pattern matching, to directly test the strictness of models in mathematical inference.
   - **Application:**

     TheoremQA is used to evaluate the capabilities of LLMs in the application of complex mathematical theorems, symbolic inference, and formula derivation.
   - **Actual scenarios:**

     TheoremQA is suitable for developing academic research tools (such as verifying solutions to math problems) and educational intelligent systems (such as advanced math tutoring), and optimizing logical inference modules (such as enhancing the symbolic computation capabilities of Mixture of Experts (MoE)-based architectures).

Knowledge Q&A

1. **MMLU**
   - **Introduction:**

     Massive Multitask Language Understanding (MMLU) is a data set covering a wide range of subjects. It contains 15,908 problems to test the interdisciplinary comprehensive understanding and inference capabilities of models.
   - **Application:**

     MMLU is used to evaluate the interdisciplinary comprehensive understanding and inference capabilities of models, covering fields such as humanity, social science, and Science, Technology, Engineering, and Mathematics (STEM).
   - **Actual scenarios:**

     In the development of intelligent assistants, MMLU can be used to train a model to answer a wide range of your questions, such as those about historical events, scientific principles,

©2013-2026 Tencent Cloud. All rights reserved.

and literary works.
In academic research, MMLU is an important tool for testing whether a model has a broad knowledge base and interdisciplinary inference capabilities.

2. **ARC-c**
   - **Introduction:**
     AI2 Reasoning Challenge - Challenge Set (ARC-c) is a science question-answering data set containing 2,590 challenging problems. It is used to test the deep inference capabilities of a model in terms of complex scientific questions.
   - **Application:**
     ARC-c is used to evaluate the deep inference capabilities of a model in solving complex scientific problems, such as high-difficulty problems in fields such as physics, chemistry, and biology.
   - **Actual scenarios:**
     In scientific research, ARC-c can be used to test whether a model can understand and solve complex scientific problems, such as explaining physical phenomena or deriving chemical reactions.
     In the education field, ARC-c can be used to develop advanced science learning tools to help students understand complex scientific concepts easily.
3. **ARC-e**
   - **Introduction:**
     AI2 Reasoning Challenge - Easy Set (ARC-e) is a simplified version of ARC-c, containing 5,197 straightforward problems. It is mainly used to test the understanding of fundamental scientific knowledge and the simple inference capabilities of models.
   - **Application:**
     ARC-e is used to evaluate the understanding of fundamental scientific knowledge and simple inference capabilities of a model, such as explaining fundamental scientific principles or answering simple science questions.
   - **Actual scenarios:**
     In the education field, ARC-e can be used to develop science learning tools for students from elementary school and middle school, helping them understand fundamental scientific knowledge.
     In the science popularization field, ARC-e can be used to train intelligent assistants to answer simple science questions posed by the public.
4. **TruthfulQA**
   - **Introduction:**
     TruthfulQA (truthful QA benchmark) comprises 817 questions designed to elicit imitative falsehoods, covering hallucination-prone fields such as health, law, and history. It can directly evaluate the truthfulness of model outputs by testing the factual accuracy and resistance to misinformation in adversarial questions of a model.

©2013-2026 Tencent Cloud. All rights reserved.

- **Application:**

  TruthfulQA is used to quantify the truthfulness of content generated by a model and its ability to identify implicit erroneous assumptions.
- **Actual scenarios:**

  TruthfulQA is suitable for developing fact-checking systems (such as automatically identifying contradictory statements), enhancing search engines (such as reliability grading), and designing safe mechanisms (such as reducing harmful, misleading outputs).

Language Understanding and Generation

1. **Hellaswag**
   - **Introduction:**

     Hellaswag is a commonsense inference data set containing 70,000 problems. It is designed to test the contextual understanding capabilities of models, particularly their comprehension and inference capabilities in daily scenarios.
   - **Application:**

     Hellaswag is used to evaluate the commonsense inference and contextual understanding capabilities of models, such as understanding the causal relationships in daily scenarios or predicting what might happen next.
   - **Actual scenarios:**

     In the development of intelligent assistants, Hellaswag can be used to train a model to better understand your daily chat, such as answering questions like What should I bring if it rains?.
     In the chatbot field, Hellaswag can be used to enhance a chatbot's decision-making capabilities in daily environments.
2. **Xsum**
   - **Introduction:**

     Xsum is an extreme summarization data set containing 226,711 news articles and their corresponding summaries. It requires a model to generate concise summaries from long texts to test its summary generation capabilities.
   - **Application:**

     Xsum is used to evaluate the capabilities of a model in generating concise summaries, and it requires a model to extract key information from long texts and generate brief summaries.
   - **Actual scenarios:**

     In the field of news media, Xsum can be used to develop automatic summarization tools to help journalists or editors quickly generate news summaries.
     In academic research, Xsum is an important benchmark for testing whether a model can accurately extract the core information from a text.
3. **TyDiQA**

©2013-2026 Tencent Cloud. All rights reserved.

- **Introduction:**

  TyDiQA is a multilingual question-answering data set containing 19,000 problems. It is used to test the multilingual reading comprehension and question answering capabilities of models.
- **Application:**

  TyDiQA is used to evaluate the capabilities of a model in multilingual reading comprehension and question answering, covering 11 languages.
- **Actual scenarios:**

  In global applications, TyDiQA can be used to develop multilingual intelligent assistants, helping you obtain information in different languages.
  In cross-cultural communication, TyDiQA can be used to train translation tools or cross-linguistic information search systems.

4. **Winogrande**
   - **Introduction:**

     Winogrande is a large-scale commonsense inference benchmark containing 44,000 pronoun resolution problems. After the Winograd Schema Challenge (WSC) is manually rewritten, Winogrande will become a larger-scale data set and can avoid bias. It focuses on testing the deep semantic understanding of entity reference relationships of a model in daily contexts, rather than superficial pattern recognition.
   - **Application:**

     Winogrande is used to evaluate the inference capabilities of a model in terms of language context, commonsense logic, and entity association.
   - **Actual scenarios:**

     Winogrande is suitable for optimizing chat systems (such as pronoun resolution and coreference modules), developing accessibility technologies (such as semantic enhancement for texts), and making research in cognitive linguistics (such as comparison between human and AI inference mechanisms).
5. **IFEval**
   - **Introduction:**

     Instruction Following Eval (IFEval) is an evaluation set that focuses on the fine-grained execution capabilities of LLMs for complex instructions. It contains over 500 human-written instructions and 25 verifiable execution criteria (for example, output strictly step by step and avoid adding explanations). It directly quantifies the adherence of a model to your explicit or implicit constraints.
   - **Application:**

     IFEval is used to evaluate the accuracy of a model in following instructions, fine-grained control capabilities, and output reliability.
   - **Actual scenarios:**

©2013-2026 Tencent Cloud. All rights reserved.

IFEval is suitable for optimizing intelligent assistants (such as accurately responding to complex requests), developing automated processes (such as precisely generating formatted outputs), and conducting safety alignment research (such as controllability verification).

Complex Inference and Comprehensive Capabilities

1. **BBH**
   - **Introduction:**

     BIG-Bench Hard (BBH) is a subset of the BIG-Bench data set, and it comprises 23 complex inference tasks that focus on testing the complex inference capabilities of models.
   - **Application:**

     BBH is used to evaluate the complex inference capabilities of models, such as logical inference, mathematical inference, and linguistic inference.
   - **Actual scenarios:**

     In AI research, BBH is an important benchmark for testing whether models have advanced inference capabilities, such as tackling logic puzzles or completing complex tasks.
     In the business field, BBH can be used to develop intelligent decision-making systems, helping enterprises analyze complex data and make informed decisions.
2. **GPQA Diamond**
   - **Introduction:**

     Graduate-Level Google-Proof Q&A Diamond (GPQA Diamond) is a benchmark data set that focuses on testing the deep inference and professional knowledge application capabilities of models on PhD-level scientific questions. It comprises 198 high-difficulty interdisciplinary questions in fields such as biology, chemistry, and physics. It is designed to evaluate whether models have complex inference capabilities similar to domain experts, rather than simple information search or recall.
   - **Application:**

     GPQA Diamond is used to evaluate the deep inference capabilities of models on PhD-level scientific questions.
   - **Actual scenarios:**

     GPQA Diamond can be used to assist in scientific research (such as answering interdisciplinary questions), develop advanced science education tools (such as testing complex concepts), and optimize models (such as enhancing the inference performance of MoE-based architectures).
3. **TEva**
   - **Introduction:**

     Text Evaluation (TEval) is a benchmark for evaluating the multi-task capabilities of Chinese LLMs, covering more than 30 task types and more than 500 fine-grained ability

©2013-2026 Tencent Cloud. All rights reserved.

dimensions (such as classical Chinese writing, legal clause analysis, and multi-hop inference). Its characteristics lie in integrating academic exam questions, professional scenario questions, and adversarial samples, providing a comprehensive evaluation of Chinese semantic understanding and generation capabilities.

- **Application:**

  TEval is used to comprehensively evaluate the performance of Chinese models in professional fields, complex inference, and cultural contexts.

- **Actual scenarios:**

  TEval is suitable for optimizing localized AI products (such as assistants in the finance or legal sectors), developing educational assessment tools (such as Chinese proficiency tests), and conducting comparative studies of multilingual models (such as analyzing differences between Chinese and English proficiency).

Code Generation

1. **HumanEval**
   - **Introduction:**

     HumanEval is a data set designed to evaluate the code generation capabilities of models, and it contains 164 programming problems.

   - **Application:**

     HumanEval is used to evaluate the code generation capabilities of models, and it requires models to generate correct code based on problem descriptions.

   - **Actual scenarios:**

     In software development, HumanEval can be used to develop automated programming tools that help programmers quickly generate code snippets.
     In the education field, HumanEval can be used to develop programming learning tools that help students practice writing code and check its correctness.

2. **MBPP**
   - **Introduction:**

     Microsoft Billion-scale Paraphrase Dataset for Programming (MBPP) is a large-scale data set of 1,000 programming problems that test the code generation and programming problem-solving capabilities of models.

   - **Application:**

     MBPP is used to evaluate the code generation and programming problem-solving capabilities of models. It covers multiple programming languages and problem types.

   - **Actual scenarios:**

     In programming education, MBPP can be used to develop intelligent programming tutoring systems that help students solve programming problems and provide feedback.
     In software development, MBPP can be used to train models to automatically generate code or optimize existing code.

©2013-2026 Tencent Cloud. All rights reserved.

Appendix

To improve the quality of model answers and parse these answers, we will add system prompts accordingly when we evaluate the following built-in evaluation sets:

- AGIEval
- ARC-c
- ARC-e
- C-Eval
- CMMLU
- Hellaswag
- MMLU

The following table shows the system prompts added based on the language types of problems:

Problem Type	Problem in Chinese	Problem in English
Single-choice question	You are an AI assistant that provides accurate answers. You are asked to answer a multiple-choice question. Please explain the question, analyze each option one by one, and then give your answer. Please end your answer with "Therefore, the answer is" followed by the letter representing the option you select.	You are a precise, user-friendly, and self-contained assistant. You are a diligent assistant. You are an error-free and error-tolerant assistant. You are a helpful AI assistant. You are an advanced, unsurpassed, and pattern-recognizing assistant. You are required to answer a single-choice question. Give your answer after you explain how to answer the question. Please reply with the ending “Therefore, the answer is” together with a letter representing your choice. You are an advanced, unsurpassed, and pattern-recognizing assistant.
Multiple-choice question	You are an AI assistant that provides accurate answers. You are asked to answer a multiple-choice	Currently, the above-mentioned evaluation sets do not include multiple-choice questions in English.

©2013-2026 Tencent Cloud. All rights reserved.

	question. Please answer directly and clearly with only one or more letters you select, without providing any explanation.		

©2013-2026 Tencent Cloud. All rights reserved.

Online Services
Introduction to Online Services

Last updated: 2026-03-03 16:53:43

Module Overview

The Online Services module of Tencent Cloud TI-ONE Platform (TI-ONE) provides the capability to deploy models as online inference services, allowing you to integrate the services with your business applications through API calls. Online Services support virtualized heterogeneous computing power and elastic scaling, helping you address issues such as complex model deployment, resource waste, and inefficient manual resource expansion. In addition, Online Services also support the deployment of multiple model formats, service traffic allocation, and rolling updates to meet diverse application requirements in online inference scenarios.

Module Features

- **Computing power virtualization**: The module supports allocating computing power as small as 0.1 GPU card for services. Through fine-grained computing power allocation, you can enjoy a cost-effective service experience anytime, anywhere.
- **Automatic elastic scaling**: You can manually or automatically adjust elastic instance scaling policies. Model deployment dynamically and automatically manages the number of instances in real-time based on business load, helping you handle business scenarios with the most suitable number of instances, and eliminating the burden of manual deployment.
- **Abundant management capabilities**: The module provides extensive support for multiple models, multi-version management, traffic allocation, rolling updates, and more. It also supports multi-dimensional monitoring of service and calling information, as well as event viewing, safeguarding your various businesses.

Scenarios

You can deploy models for various machine learning scenarios, including recommendation, image processing, natural language processing, and automatic speech recognition, as Online Services.

©2013-2026 Tencent Cloud. All rights reserved.

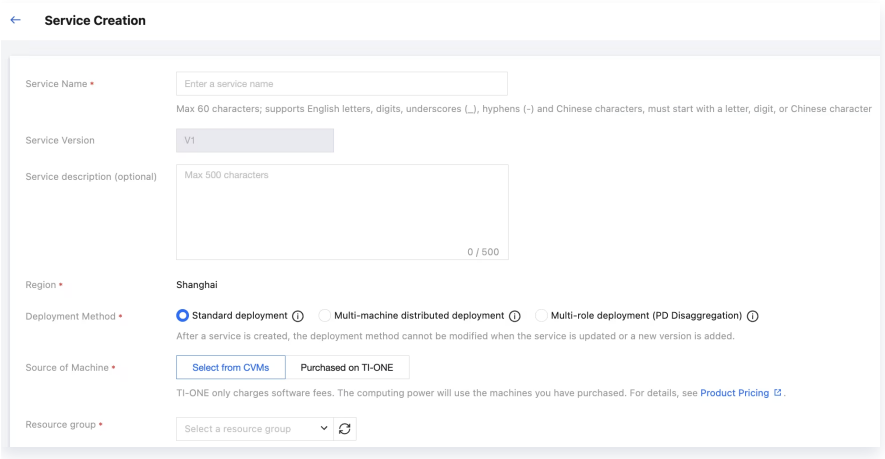
Deployment of Online Services

Last updated: 2026-03-12 10:41:28

Basic Information Configuration

Log in to TI-ONE console, choose **Model Service > Online Services** in the left sidebar to go to the Online Services list page. On the service list page, click **Create Service** to go to the service startup page. On the service startup page, configure the relevant parameters for the online services.

1. Basic service information



Parameter description:

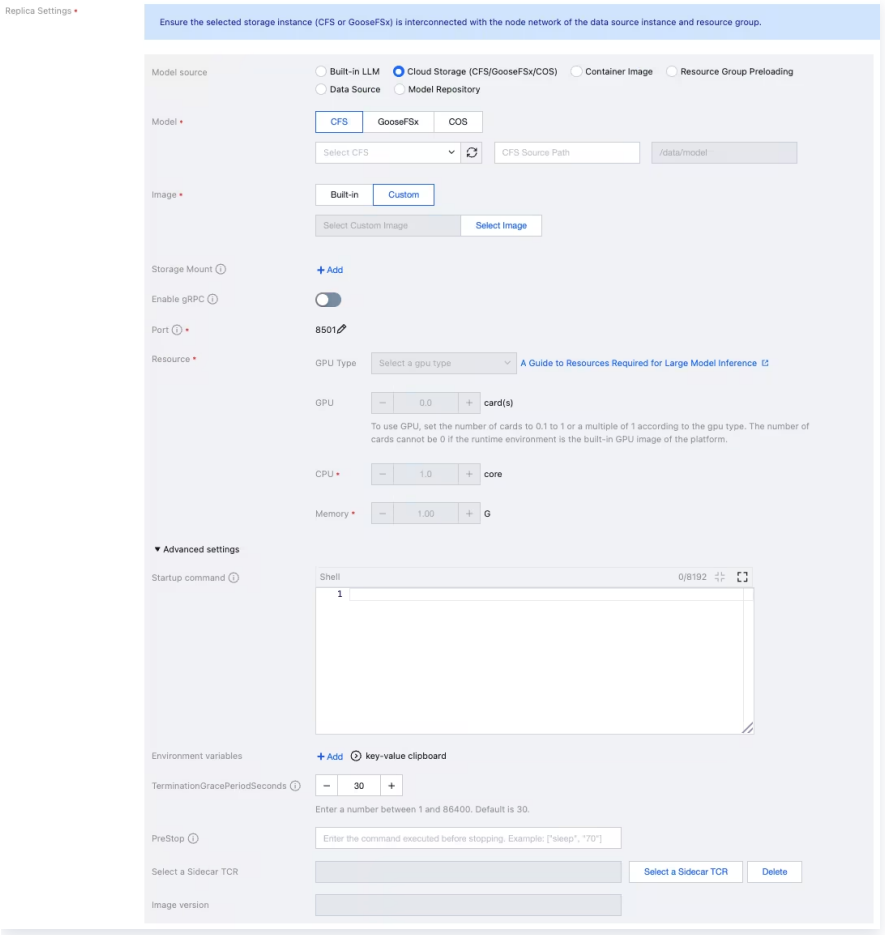
Parameter	Description
Service Name	Name of the service. You can enter it based on the rules prompted on the interface.
Service Version	Version number automatically generated by the system.
Service Description	Configure a description for the service as needed.
Region	Services under the same account are isolated by region. The value of the Region field is automatically entered based on the region you selected on the service list page.
Deployment Mode	Multiple deployment modes are supported: • Standard deployment: 1 instance runs under a single replica, suitable for most standard scenarios.

©2013-2026 Tencent Cloud. All rights reserved.

	• Multi-machine distributed deployment: Multiple instances run in coordination under a single replica, suitable for scenarios where models require multi-machine parallelism. • Multi-role deployment (prefill and decode separation): Currently only available through allowlist; suitable for scenarios where the service needs to be deployed with multiple roles (such as prefill and decode roles). There are multiple roles under a single replica, and each role has 1 or more instances running in coordination. **⚠ Note:** After a service is created, the deployment mode cannot be modified by updating the service or adding versions. Choose carefully during creation.
Source of Machine	You can choose Select from CVMs instances or Purchased on TI-ONE : • In the Select from CVMs instances mode, you can use resource groups of CVM instances purchased in the Resource Group Management module to deploy services. The computing resource fee has been paid when the resource group is purchased, so no fees will be deducted when the service is started. • In the Purchased on TI-ONE mode, you do not need to purchase a resource group in advance. Fees are charged based on the CVM instance specifications required by the service. When the service is started, fees for the first two hours are frozen. After that, fees are charged hourly based on the number of running instances.
Resource Group	If you choose the Select from CVMs instances mode, you can select resource groups from the Resource Group Management module.

Service Replica Configuration

©2013-2026 Tencent Cloud. All rights reserved.



Parameter description:

Parameter	Description
Model Source/Model	Multiple model sources are supported (the supported model sources may not be entirely consistent across different deployment modes): • Built-in LLM: industry-leading open-source LLMs provided in the platform's Model Hub. • Cloud Storage (CFS/GooseFSx/COS) ○ CFS: The model files required for service deployment are stored in Cloud File Storage (CFS). Select the CFS instance where the model is located. When entering the path, specify it down to the level of the model's directory (for example, if the model is a fine-tuned checkpoint500, enter /a/b/checkpoint500 as the path). CFS Turbo is only supported when the CVM instance source is Select from CVM instances. ○ GooseFSx: The model files required for service deployment are stored in GooseFSx. Select the GooseFSx instance where the model is located, and enter the directory of the model in the path field. GooseFSx is only supported when the CVM instance source is Select from CVM instances.

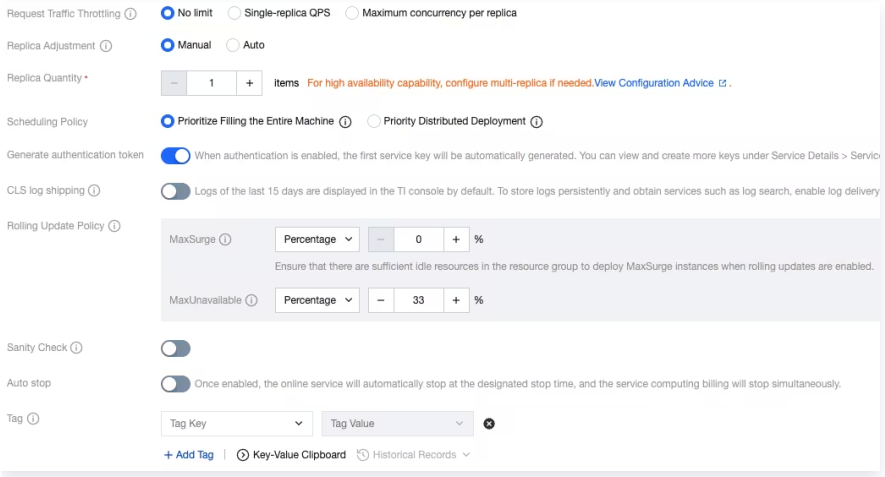
©2013-2026 Tencent Cloud. All rights reserved.

	○ COS: The model files required for service deployment are stored in Cloud Object Storage (COS). Select the bucket instance where the model is located and enter the directory. COS is only supported when the CVM instance source is Select from CVM instances. • Container image: The custom image required for service deployment has encapsulated the model files, eliminating the need for model file mounting, and the custom image has been uploaded to Tencent Container Registry (TCR). • Resource Group Preloading: Only applicable to the scenario of Select from CVM instances. You can preload model files/image files to the corresponding resource group in advance, improving the model loading speed during service startup. • Data Source: Select the data source file that you have pre-added in the Platform Management - Data Source Management module. The Data Source Management module provides unified file permission management. • Model repository: The model files required for service deployment have been imported into the Model Repository module.
Image	• Built-in LLM: The platform provides a corresponding built-in running image for each built-in LLM; you do not need to modify it. • Cloud Storage/Resource Group Preloading/Data Source: All support 2 types of image selection. ○ Built-in: The platform provides various open-source built-in images and Tencent's self-developed images for inference acceleration. ○ Custom: You can choose TCR or enter a custom image address (along with the username and password for the private image repository, if applicable). • Container image: You can directly select a custom image that has been uploaded to TCR, and the image contains the required model files. • Model Repository: The running environment will automatically assign values based on the configuration information of the model repository. If you need to use a custom image to start the service, it is recommended that the image file size does not exceed 34 GB.
Storage Mount	When creating an online service, users are supported to configure file mount paths for input and output while also configuring model files.
Enable gRPC	The switch is disabled by default. When the switch is disabled, only calls using HTTP are supported. When the switch is enabled, calls using the gRPC protocol are supported.
Port	You can configure the ports exposed by the container. The valid port range is 1024-65535, excluding 8502-8510, 6006, and 9092.

©2013-2026 Tencent Cloud. All rights reserved.

	Pay special attention to two special port numbers: 8500 is the default port for gRPC, and 8501 is the default port for REST; they must not be confused.
Resource	1. In the yearly/monthly subscription (resource group) mode, you can set how many resources to request from the selected resource group to start the current service. 2. In the pay-as-you-go mode, you can choose the CVM instance specifications required for starting the current service as needed.
Startup Command	Optional. You can configure the startup command of a container.
Environment Variable	Optional. You can configure the environment variables of a container.
Graceful Shutdown Period	Corresponds to Kubernetes' terminationGracePeriodSeconds. Pods whose shutdown time exceeds this limit will be forcibly terminated. The default value is 30s.
PreStop	Optional. Corresponds to Kubernetes' PreStop command. Pods run this command before termination to achieve a graceful shutdown. The command format is a string array, for example: "["sleep", "70"]".
Select a Sidecar TCR	Optional. You can customize the sidecar container image.

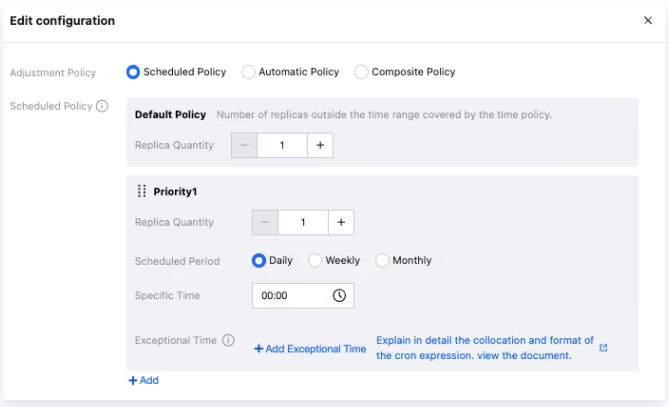
Service Feature Configuration



Parameter description:

Parameter	Description
Request Traffic	You can configure traffic throttling values for the service:

©2013-2026 Tencent Cloud. All rights reserved.

Throttling	• When traffic throttling is not applied, the default maximum queries per second (QPS) per service is 500. If the configured traffic throttling value exceeds 500, it will be capped at 500. After an upgrade package is purchased for the service, the total traffic throttling value of the service is subject to the upper limit of the purchased upgrade package. • The maximum QPS per replica is the traffic throttling for one instance. • The maximum concurrency per replica is the traffic throttling for one instance. **Note:** This traffic throttling value applies to a single replica. When the service scales, the overall traffic throttling value of the service is updated according to the set value multiplied by the number of replicas. The maximum QPS for a single service group is 500. If the total traffic throttling value set for the services under the service group exceeds 500, it will be capped at 500.
Replica Adjustment	• Manual: You can customize the number of service replicas, with a minimum value of 1. • Auto: You can choose a timed policy, an HPA-based automatic policy, or a combined timed+HPA policy. For details, please refer to Online Service Operation. 
Use RDMA	When the number of replicas or the number of instances per replica exceeds 1, and the selected GPU model supports RDMA, you can enable RDMA. After RDMA is enabled, services are preferentially scheduled to nodes supporting RDMA. If a single node lacks sufficient IP addresses for RDMA NICs, services may be scheduled to multiple nodes.
Scheduling Policy	For the "Standard Deployment" mode, you can customize the node scheduling policy for multiple replicas: • Prioritize Filling the Entire Machine: When multiple replicas are scheduled, prioritize packing them onto a single node to reduce GPU

©2013-2026 Tencent Cloud. All rights reserved.

	fragmentation probability. • Priority Distributed Deployment: When multiple replicas are scheduled, prioritize distributing them across different nodes to enhance service high availability (note: successful scattered deployment requires sufficient resources).
Generate authentication token	If authentication is enabled, signature authentication will be performed during service calling. For started services, you can view the signature key and signature calculation guide on the service calling page. After authentication is enabled, the first key for the service will be automatically generated. You can view and create additional keys under Service Authentication on the service details page.
CLS Log Shipping	The platform provides free storage of service logs for the last 15 days. If you require persistent log storage, more flexible log search capabilities, and log monitoring alarm capabilities, you can enable CLS Log Shipping, and then service logs will be shipped to Cloud Log Service (CLS) according to the logset and log topic.
Rolling Update Policy	You can configure the rolling update policy to ensure smooth service upgrades: • MaxSurge: maximum number of extra replicas that can be created beyond the desired number during a rolling update. • MaxUnavailable: upper limit of the number of unavailable replicas allowed during a rolling update.
Sanity Check	The health check mechanism of Kubernetes automatically detects and recovers failed containers, ensuring traffic is routed to healthy instances. • Liveness probe: verify whether the service process is alive and whether the container is running normally. Trigger phase: continuously run after container startup (throughout the entire lifecycle). • Readiness probe: verify whether the service is ready to process requests and whether the container is prepared to receive traffic. Trigger phase: continuously run after container startup (throughout the entire lifecycle). • Startup probe: monitor slow-startup containers and verify if the application within the container has completed initialization. Trigger phase: run only during container startup (stop after success). Three check methods are supported: HTTPGet, TCPSocket, and Exec.
Auto stop	The platform supports the automatic stop of model services. After Automatic Stop is enabled, online services will stop automatically at the specified stop time, and the computing power billing for the service will also stop.
Tag	You can add tags to services for authorization based on tags.

©2013-2026 Tencent Cloud. All rights reserved.

After confirming that the configuration information of the service is correct, click **Start Service** to deploy the service. During deployment, a gateway will be created, and computing resources will be scheduled for you. This process takes some time. Once the service is successfully deployed, its status will change to **Running**.

Deployment Parameters Best Practices

This section provides a detailed explanation of some important parameters for service deployment and offers best practice recommendations for parameter configuration.

Health Check

1. What Are Health Checks

Health checks are a critical mechanism to ensure the stable operation of your online services. By periodically checking the health status of service instances (Pods), they ensure that traffic is only routed to healthy instances. Unhealthy instances are restarted or replaced, thereby enhancing the service's availability and reliability.
TI-ONE provides three types of health check probes based on Kubernetes:

Probe Types	Startup Probe	Readiness Probe	Liveness Probe
Core role	Startup probes verify whether the application within the container has completed initialization.	Check whether the container is ready to receive traffic.	Check whether the container is running properly.
Trigger phase	Runs only during the container startup phase (stops upon success).	Continuously runs after container startup (throughout the entire lifecycle).	Continuously runs after container startup (throughout the entire lifecycle).
Failure Consequence	Terminate the container and restart it.	Removed from the service CLB pool, traffic forwarding stopped (without restarting the container).	Terminate the container and restart it.
Typical Scenario	Slow startup applications (such as Java service warm-up and data loading)	Dependencies initialization completed (such as database connections and configuration file loading)	Handle deadlocks, process crashes, or unrecoverable exceptions.

©2013-2026 Tencent Cloud. All rights reserved.

Detection Frequency	Low frequency (usually with longer intervals, such as 10s)	Medium frequency (adjustable based on business requirements)	High frequency (quickly detect abnormalities, such as 5s)
Priority	Highest (disable other probes during startup)	Medium	Low

2. The Role of Health Checks

- Improve service availability

Through readiness probes, ensure that only prepared instances receive traffic, preventing requests from being sent to instances that are not ready yet, thereby reducing request failures to zero.

- Automatically recover from failures

Through liveness probes, when a failure in a running instance is detected (such as application deadlock but process still running), the system automatically restarts the instance to rapidly restore service.

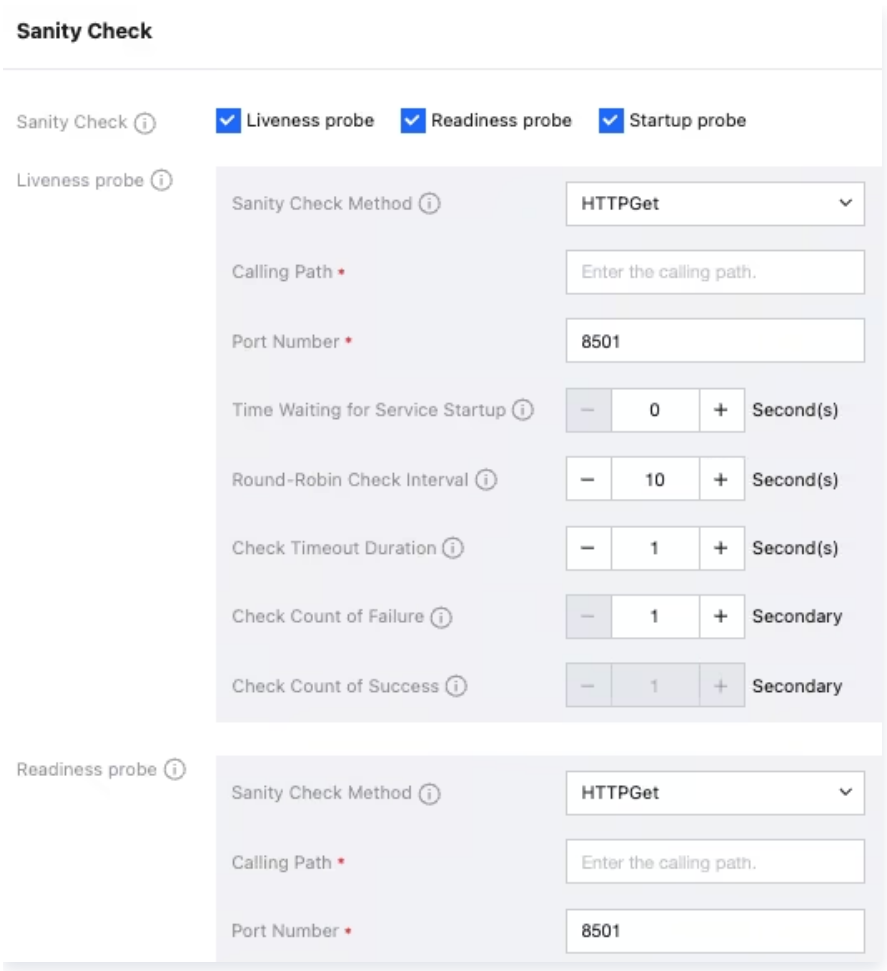
- Graceful handling of startup and termination

For applications with slow startup, readiness probes prevent traffic reception during initialization; for applications requiring graceful shutdown, readiness probes coordinate with lifecycle hooks to ensure removal from load balancers before termination.

3. How to Configure Health Checks

On the TI-ONE platform, you can configure health checks in the parameter form when creating or updating a service. You will need to select the appropriate detection method based on your application characteristics.

©2013-2026 Tencent Cloud. All rights reserved.



3.1 Detection Method

TI-ONE supports the following three detection methods:

Detection Method	Method Description	Scenarios	Configuration Example
HTTP GET Probe	Send an HTTP GET request to the specified path and port within the container. If the returned status code is between 200 and 399, it is considered successful.	Web services, API services	Path: /health, Port: 8080
TCP Socket Probe	Attempt to establish a TCP connection with the container's specified port. If the connection is successfully established, it is considered successful.	Databases, caches, non-HTTP services	Port: 6379 (Redis)

©2013-2026 Tencent Cloud. All rights reserved.

Exec Command Execution Probe	Execute the specified command within the container. If the command's exit code is 0, it is considered successful.	Services requiring complex check logic	Command: python check_health.py

3.2 Configuration Parameter Description

The supported configuration parameters for the three types of probes are described as follows:

Parameter	Parameter Description	Recommended Values for Parameters
Delay for waiting service startup	After the container is started up, it indicates how long to wait before initiating the first probe. Set this value based on the application startup time to avoid false negatives due to the application not being fully started.	For example: 30-60 seconds, to give the application sufficient startup time and avoid false negatives.
Polling Probe Interval	The interval between each probe. Too high a frequency imposes significant overhead on Pods, while too low a frequency fails to promptly reflect container failures.	For example: 5-15 seconds, to balance real-time requirements and system overhead.
Check Timeout Duration	Timeout Duration per Probe, beyond which the probe is considered failed.	For example: 3-5 seconds, to prevent false negatives caused by network jitter.
Failure Threshold	Number of Consecutive Failures Before Marking Instance as Unhealthy	For example: 3 times, marking as unhealthy only after 3 consecutive failures to avoid false positives from transient failures.
Success Threshold	Consecutive Number of Successes Before the Instance is Marked as Healthy	For example: 1 time. Setting it to 1 for transient failures that can self-heal quickly enables prompt service recovery.

3.3 Configuration Example

Suppose you have a Web service listening on port 8080 with a health check endpoint at /health. This service takes approximately 30 seconds to start.

- Liveness Probe Configuration (Recommended):
  - Check Method: HTTP GET

©2013-2026 Tencent Cloud. All rights reserved.

- Invocation Path: /health
- Port: 8080
- Delay for waiting service startup: 40 seconds (ensuring the application has started)
- Polling Probe Interval: 10 seconds
- Check Timeout Duration: 5 seconds
- Failure Threshold: 3
- Success Threshold: 1

- Readiness Probe Configuration (Recommended):
  - Check Method: HTTP GET
  - Invocation Path: /health
  - Port: 8080
  - Delay for waiting service startup: 40 seconds
  - Polling Interval: 5 seconds (Readiness checks can be more frequent to allow quick addition to CLB)
  - Check Timeout Duration: 5 seconds
  - Failure Threshold: 1 (Marked as not ready after a single failure)
  - Success Threshold: 1

> ⚠ **Note:**
> If your service does not have a dedicated health check interface, you can use the root path / or perform a TCP port check. For non-Web services, you may choose TCP checks or command execution checks.

4. How to View Health Check Status

On the TI-ONE platform, you can view the status and events of health checks by:
4.1 Service Instance List
Go to the online service details page and view the "Instance List" Tab page. Each instance displays its current status (Running, Ready, and so on). If an instance restarts due to health check failures, you can see the number of container restarts and status changes.
4.2 Event Log
Go to the online service details page and view the "Events" Tab page. The platform will record events such as health check failures and container restarts. Here you can see specific failure causes, such as "Liveness probe failed" or "Readiness probe failed".

©2013-2026 Tencent Cloud. All rights reserved.

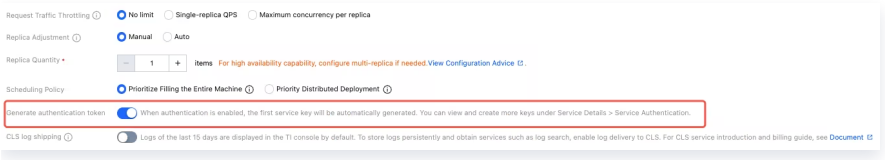
Authentication and Traffic Throttling for Online Services

Last updated: 2026-03-03 16:55:23

The Online Services module of Tencent Cloud TI-ONE Platform (TI-ONE) provides you with authentication and traffic throttling for LLM services. It allows you to configure multiple keys for a single service and supports traffic control for LLMs based on token counting, thereby achieving precise caller management and traffic control.

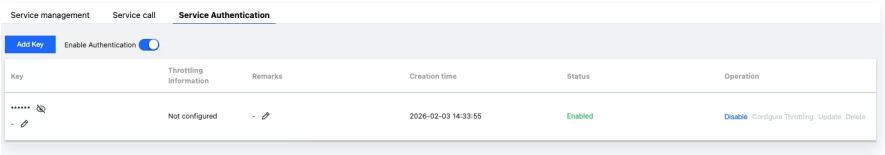
Enabling Service Authentication

You can enable the **Whether to Generate Authentication** switch on the parameter configuration page when you create an online service to enable the service authentication feature. Since the status of the authentication switch is independent of the deployed service, after the service is created, you can change the status of the authentication switch by editing it in the Service Authentication tab on the service details page without stopping or updating the service.



Managing Authentication Keys

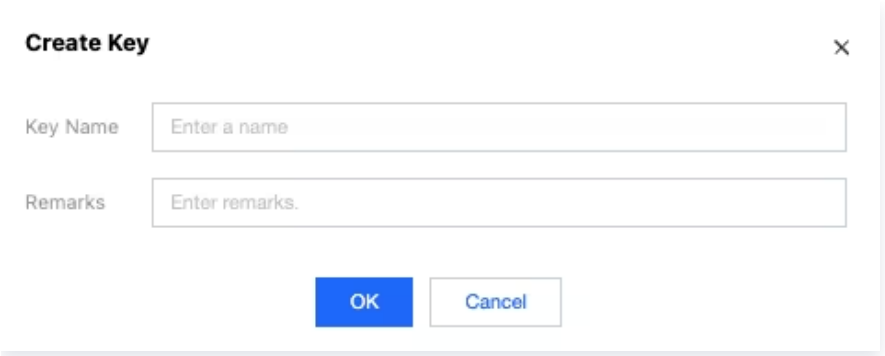
After enabling authentication keys, you can click the online service name to go to the service details page and centrally manage authentication keys through the **Service Authentication** tab.



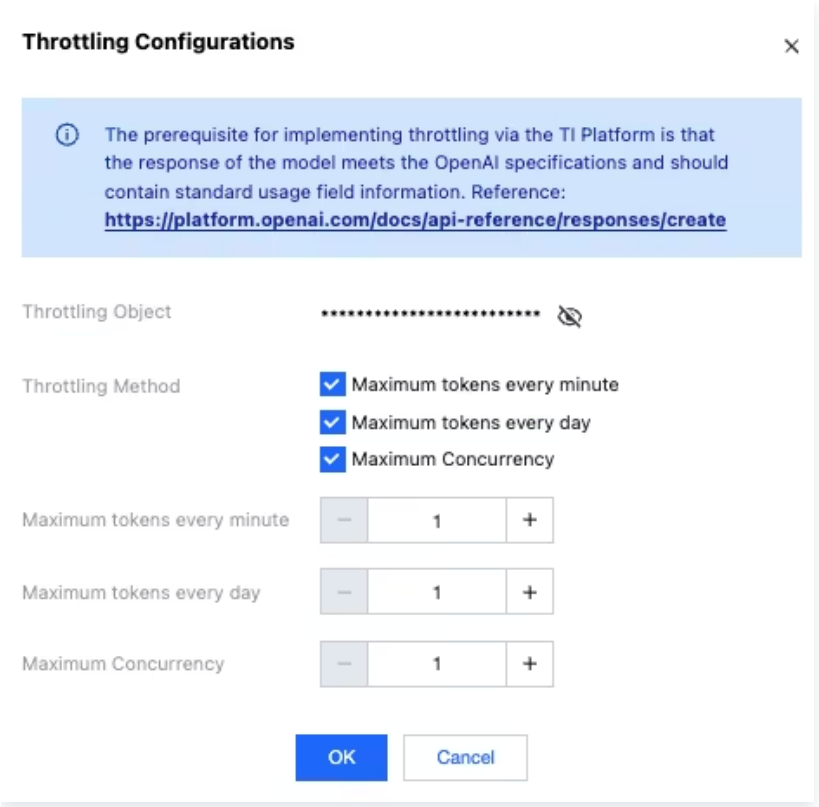
The fields in the list on this page are described as follows:

- Key: displays the key information required for service access. Currently, only the key type of AuthToken is supported. After authentication is enabled, you need to carry this AuthToken in service requests for authentication. You can click **Add Key** on the list page to add a custom key.

©2013-2026 Tencent Cloud. All rights reserved.



- Traffic throttling information: displays the traffic throttling settings you configured for each key. 3 types of traffic throttling are supported: Maximum tokens every minute (TPM), Maximum tokens every day (TPD), and Maximum Concurrency.



- Remarks: display remarks you added for keys, allowing you to record the purpose of each key in detail.
- Creation time: Time when the key was created.
- Status: Current status of the key, with two enumerated values: Enabled/Disabled. You can choose **Actions > Disable/Enable** to change the key status. After a key is disabled, TI-ONE will reject all service requests using this key. Proceed with caution.

Service Traffic Throttling

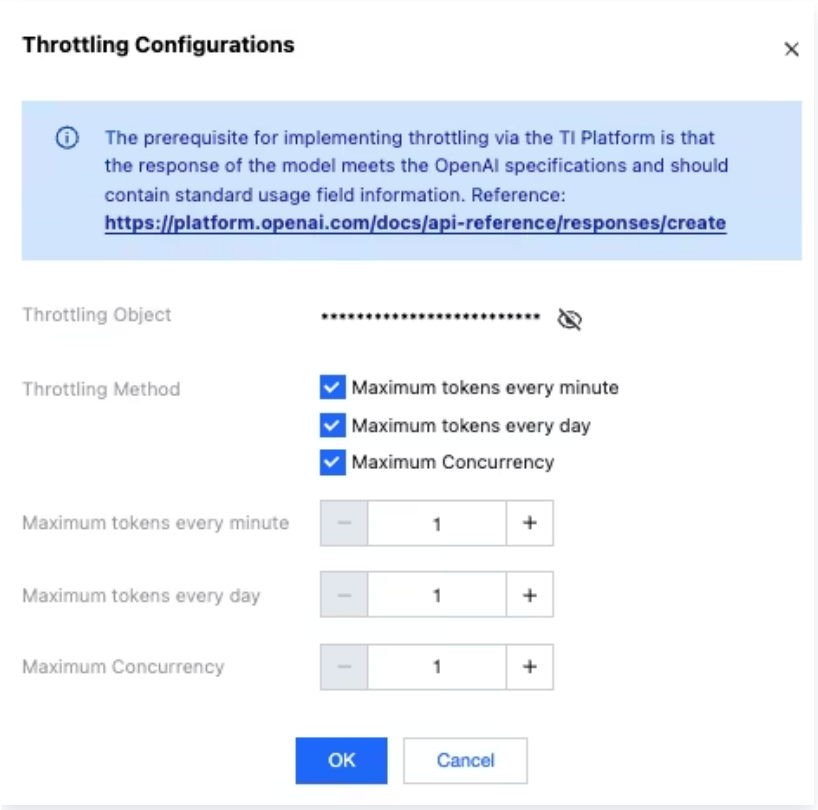
You can set traffic control for corresponding keys by clicking Operation - Throttling on the service authentication list page. Since non-LLM services lack the concept of tokens, token-based traffic throttling is unavailable. Therefore, the traffic throttling feature only supports LLM

©2013-2026 Tencent Cloud. All rights reserved.

services where the Model Source is Image-Built-in LLMs or the Running Environment is Built-in LLMs. A prerequisite is that the model's response must comply with the OpenAI specification and include standard information of the usage field.
Currently, 3 traffic throttling methods are supported:

- **Maximum tokens every minute**: The maximum number of tokens that can be requested per minute using this key.
- **Maximum tokens every day**: The maximum number of tokens that can be requested per day using this key.
- **Maximum Concurrency**: Note that the Maximum Concurrency configured here is refined traffic throttling at the key level. If you have also set the "Maximum Concurrency per Replica" at the service level when creating the service, your requests will also be subject to the global concurrency restriction of service replicas.

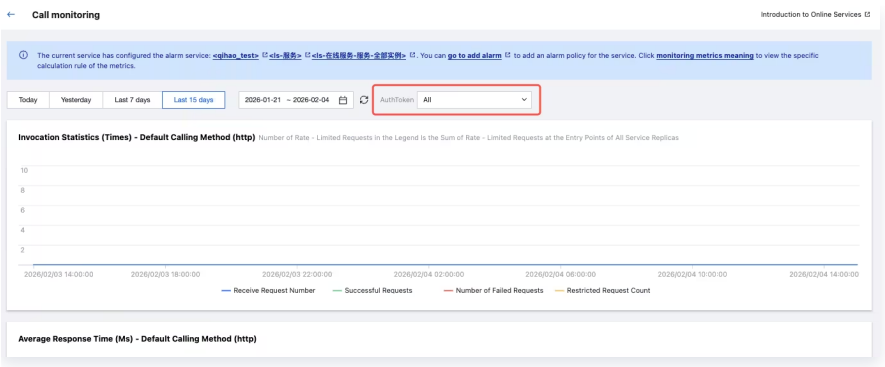
After traffic throttling is set for a key, if service requests using this key exceed the traffic limit, the requests will fail.



Monitoring of Service Authentication Calls

Since the platform allows you to configure multiple authentication keys for a single LLM service, on the call monitoring page for LLM services, the platform also allows you to view detailed call information for each caller by switching authentication AuthTokens, achieving precise traffic monitoring.

©2013-2026 Tencent Cloud. All rights reserved.



©2013-2026 Tencent Cloud. All rights reserved.

Operations of Online Services

Last updated: 2026-03-03 16:55:45

Automatic Scaling

If your business workload has distinct peak and off-peak characteristics, you can use the auto scaling capability of the online service module to improve the utilization efficiency of inference computing resources. This feature supports automatically adjusting the number of online service instances based on your configured scaling policy, thereby achieving automatic scale-out during business load peaks and automatic scale-in during business load troughs.
Auto scaling supports multiple types of adjustment policies: time-based adjustment, HPA-based adjustment, and combined time+HPA adjustment. The following details the method of use for different adjustment policies.

Time-Based Adjustment (Timed Policy)

If your workload has distinct time characteristics, you can configure the automatic scaling policy based on the specific time.

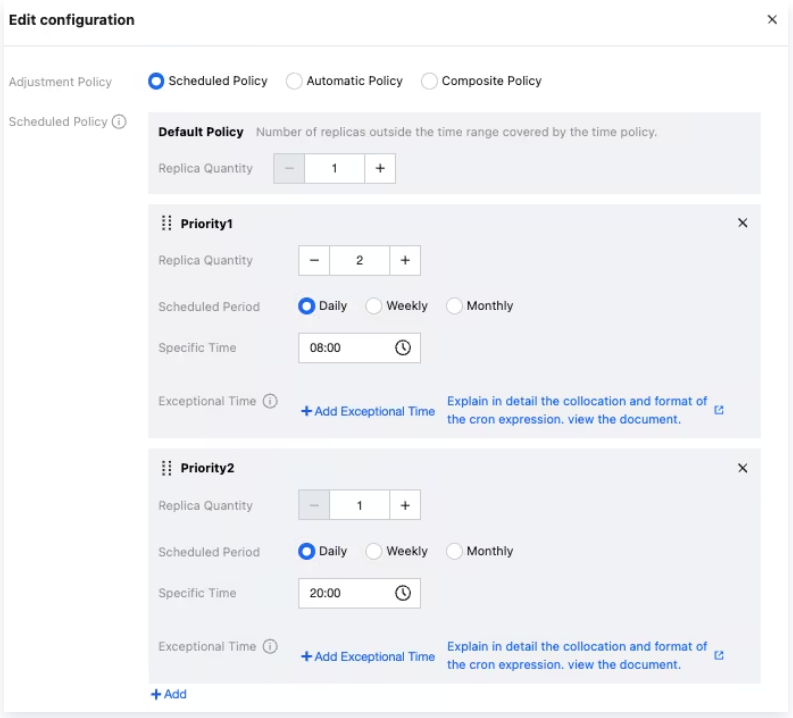
1. **Configure timed policy.**

   1.1 Log in to the Product Console, choose **Model Service** > **Online Service** in the left sidebar, click the [New Service] button to enter the parameter configuration form, configure the "Replica Adjustment - Auto" parameters, and click the "Edit Configuration" button.



   1.2 In the Edit configuration window, select "Adjustment Policy: Scheduled Policy" to configure rules for the time-based scaling policy. You can configure multiple timed policy rules based on the time characteristics of your actual workload. For example, if the peak hours range from 8:00 AM to 8:00 PM and the off-peak hours range from 8:00 PM to 8:00 AM, you can configure the timed policy as shown in the figure below. This policy scales out instances to 2 at 8:00 AM every day and scales in instances to 1 at 8:00 PM every day (the default policy uses the initial number of instances after the service is started).

©2013-2026 Tencent Cloud. All rights reserved.



1.3 If you have configured multiple timed policy rules and a time conflict exists between these rules, the policy with a higher priority (that is, the one ranked higher) will prevail.

2. **Exceptional time configuration rules**

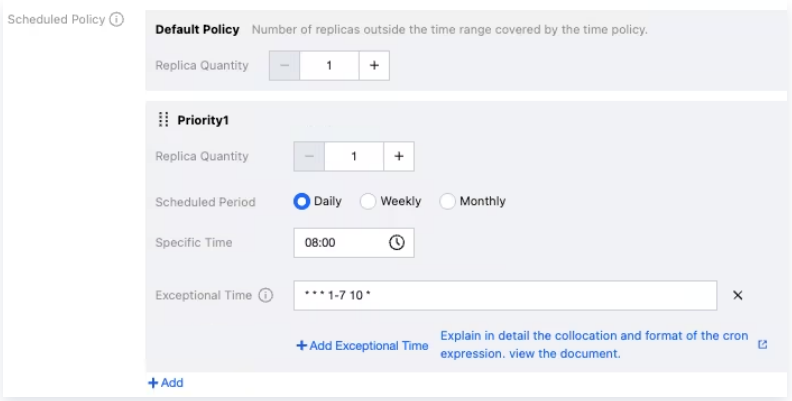
2.1 If you want a timed policy not to be executed at a specific time, you can configure an exceptional time value for this policy rule. Multiple exceptional time values can be added.

2.2 An exceptional time value needs to be configured through a Cron expression. A Cron expression consists of 6 digits, representing second, minute, hour, day, month, and week respectively. If the value of a specific digit is any value, an asterisk (*) is used. If the value of a specific digit needs to contain multiple consecutive values, a hyphen (-) is used. If the value of a specific digit needs to contain multiple discrete values, a comma (,) is used.

2.3 The minimum granularity configured for an exceptional time value is in days. Therefore, the first three digits of a Cron expression need to use "*" (other values configured for the first three digits will not take effect), and the last three digits can be configured as needed. The available value ranges from 1 to 31 for the 4th digit "day", 1 to 12 or January to December for the 5th digit "month", and 0 to 6 or Sunday to Saturday for the 6th digit "week".

2.4 Example: The Cron expression for October 1 to October 7 every year is "* * * 1-7 10 *".

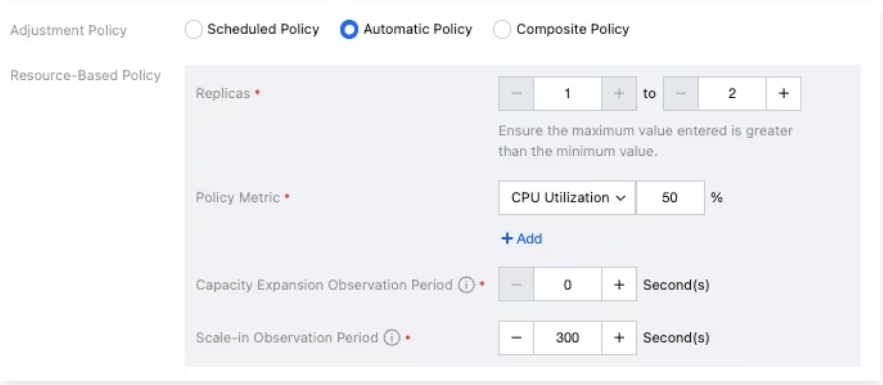
©2013-2026 Tencent Cloud. All rights reserved.



HPA-Based Adjustment (Automatic Policy)

If a timed policy does not suit your business model, you can also choose "Adjustment Policy: Automatic Policy".
Under this policy, the number of service instances can be automatically adjusted between the minimum and maximum values based on the policy metrics and thresholds you configured, and supports user configuration of:

- Policy metrics: Supports metrics such as CPU utilization, memory utilization, GPU utilization, QPS per replica, number of requests in processing, number of input tokens processed, number of output tokens processed, token utilization, and so on
- Scale-out observation period: During the observation period, a scale-out operation is performed only when all sampled values exceed the preset threshold.
- Scale-in observation period: During the observation period, a scale-in operation is performed only when all sampled values are below the preset threshold.



Timed+HPA (Composite Policy)

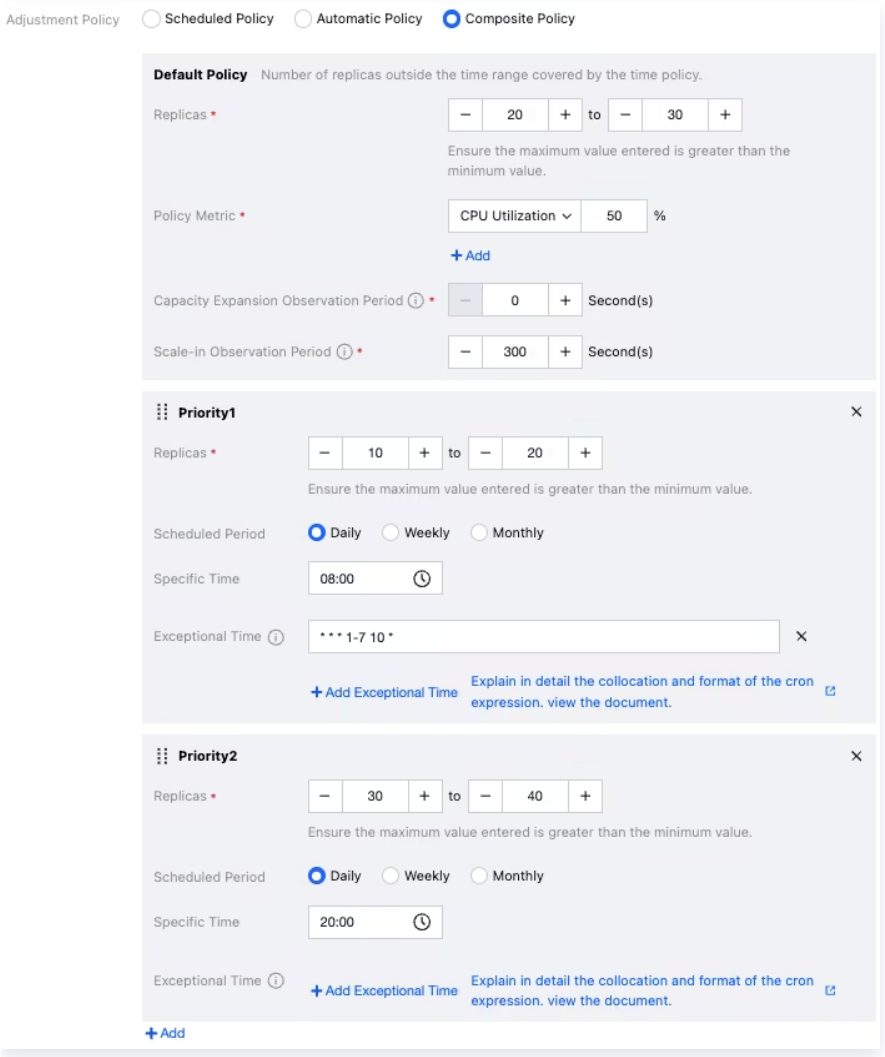
Scheduled scaling and HPA auto scaling are two different elastic scaling policies in Kubernetes, with significant differences in trigger mechanisms, applicable scenarios, and configuration methods. The hybrid policy of scheduled + HPA can fully leverage the advantages of both mechanisms, ensuring service stability while achieving cost optimization and improved resource utilization.

©2013-2026 Tencent Cloud. All rights reserved.

As follows, a real business scenario requires configuring a combined scaling policy:

- First, specify a default global scaling policy as follows: replica range: 20–30, CPU utilization metric: 50%;
- Then, at specific time point 1: 8:00 AM is the off-peak period. Starting at 8:00 AM, the scaling policy is set to: replica range: 10–20, CPU utilization metric: 50%.
- Then, at specific time point 2: 20:00 is the peak period. Starting at 20:00, the scaling policy is set to: replica range: 30–40, CPU utilization metric: 50%.

The combined policy configuration for this scenario is as follows:



Traffic Allocation

To meet the requirements of grayscale verification or A/B testing services, the platform allows you to add multiple versions for a single service and allocate traffic.

1. Log in to the Product Console, choose **Model Service** > **Online Service** in the left sidebar to go to the Online Services list page.

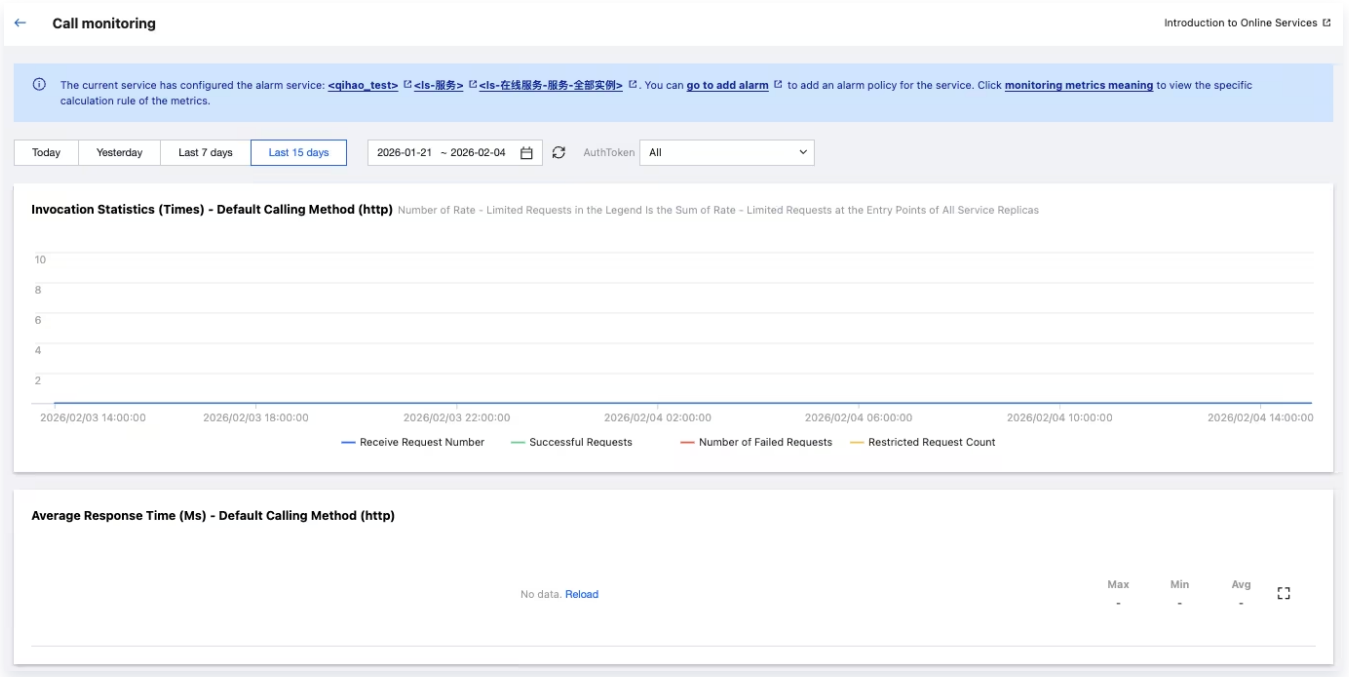
©2013–2026 Tencent Cloud. All rights reserved.

2. Locate the service to be tested, click Add **Version** to go to the service version creation page, and then configure the container information and instance scaling information of the current service version as needed.
3. Click **Start Service**. If the postpaid billing mode is used, you need to confirm the freezing of the fees to complete the version creation.
4. After you create a service version, the system will create a gateway backend and schedule computing resources for you, which takes some time. When the service version is successfully deployed, the status will become **Running**.
5. Click **Traffic Allocation** above the service version list to configure the traffic proportion of multiple versions.

Monitoring Services

To meet the requirements for tracing the running status of services, the platform provides capabilities such as service data monitoring, calling data monitoring, and event and log viewing.

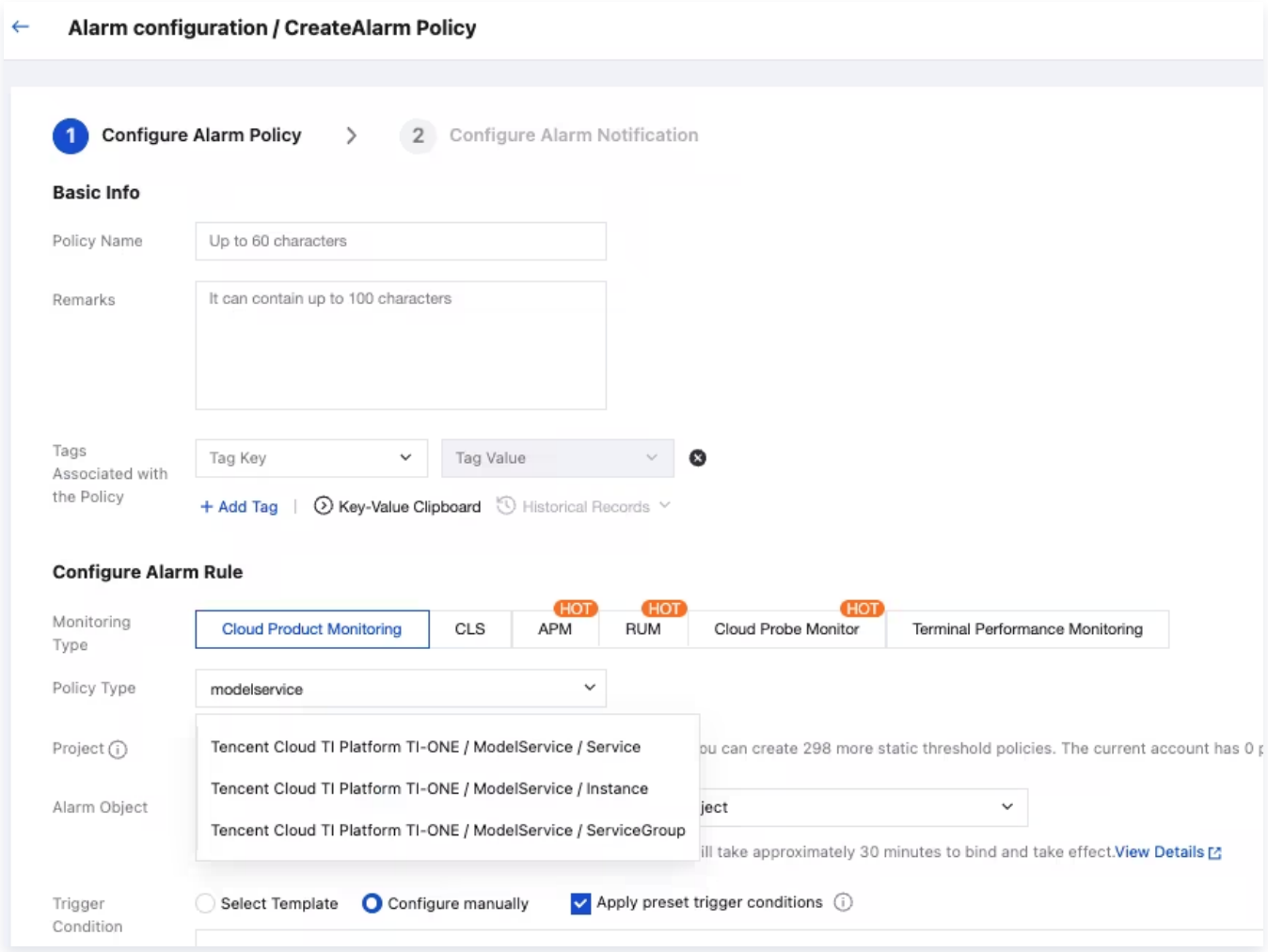
1. On the Online Services list page, click the **service name** to go to the version list page. Then choose **Service Call** > **Call Monitoring** to view the statistical information of the service calling status, including the number of received requests, number of successful requests, number of failed requests, number of restricted requests, and average response time.



2. On the monitoring page of the Online Services module, you can redirect to the **Alarm Management** page in **Tencent Cloud Observability Platform (TCOP)** to add an alarm policy for

©2013-2026 Tencent Cloud. All rights reserved.

the service.



3. On the Online Services list page, click the **service name** to go to the version list page. Then click **Service Version Name** to go to the version details page. You can view service monitoring, event monitoring, and running logs. Service monitoring is divided into LLM monitoring and general monitoring:
   - LLM monitoring

Category	Monitoring Metric	Monitoring Metric Description
Latency	TTFT (s)	Time to first token (TTFT) refers to the time taken between when you initiate a request and when the model starts to output the first token. In streaming output scenarios, this metric determines when the content begins to be updated on the user interface, and it is critical for ensuring a smooth user experience.
	TPOT (s)	Time per output token (TPOT) measures the average time taken to generate each subsequent

©2013-2026 Tencent Cloud. All rights reserved.

		token once the model produces the first output token. It focuses on the efficiency of the decoding phase and reflects the autoregressive text generation capability of the model. OTPS = 1/TPOT, indicating that the two metrics are reciprocals of each other.
	TRT (s)	Total response time (TRT) measures the total time it takes to initiate a request and receive a response, that is, the time from when a client initiates a request until it receives the response from the server. It is also known as the response time or request processing time.
Throughput	RPM	Requests per minute (RPM) measures the number of requests the Model Services module can process within a minute. For example, if you set the RPM of the model to 300, it can respond to a maximum of 300 user requests per minute.
	TPM	Tokens per minute (TPM) measures the total number of input and output tokens that an LLM can process per minute. You can view TPM for Input and TPM for Output.
	System average output tokens (token/s)	Output tokens per second (OTPS) represents the number of output tokens generated by the model per second. It is used to evaluate the overall efficiency of the model in generating text. Higher OTPS values represent smoother model output and a more seamless user experience.
Token accumulation and characteristics	Total number of tokens processed	Total length of input tokens and output tokens throughput a day. You can view Total Input Tokens and Total Output Tokens.
	Average input length (token)	Average length of all input samples.
	Average length of output (token)	Average length of all output samples.
Request load	Number of requests in processing	Number of requests in processing during sampling.

©2013-2026 Tencent Cloud. All rights reserved.

	Number of queued requests	Number of requests in the queue during sampling. It is usually used when no more tokens can be stored in the KV cache.

- General monitoring

Category	Monitoring Metric	Monitoring Metric Description
Traffic information	Inbound network traffic	Downstream (ingress) traffic for network data transmission of a service instance. It reflects the communication load of the Model Services module. Burst traffic may indicate a surge in requests or an abnormal call, and it can be analyzed together with QPS.
	QPS (times/second)	Number of requests processed per second (queries per second). It is a core metric for measuring service throughput.
	QPS traffic throttling (times/second)	A mechanism that actively limits the maximum requests per second. If the allowed number of requests is exceeded, the additional requests will be rejected.
	Number of concurrent requests	Number of requests being processed at the same time (including requests waiting to be processed and those being calculated).
Resource information	CPU utilization (%)	CPU usage percentage of a service instance.
	MEM utilization (%)	Memory usage percentage of a service instance.
	GPU memory utilization (%)	GPU memory usage percentage of a service instance.
	GPU utilization (%)	Load percentage of GPU computing units of a service instance.
Instance information	Number of instances	Total number of configured service instances (including those in all states such as Starting, Running, and Abnormal).
	Number of running instances	Number of instances that actually process requests normally.

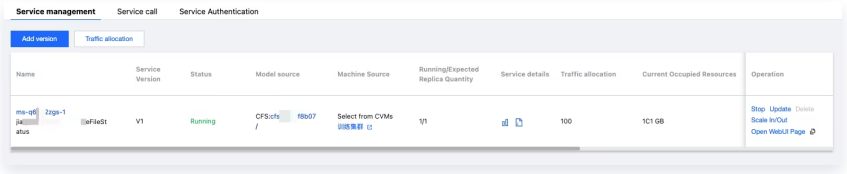
©2013-2026 Tencent Cloud. All rights reserved.

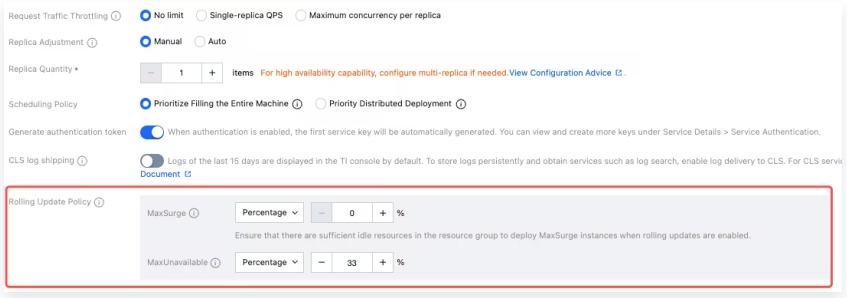
Updating Services

Choose Operation > **Update** to update the configurations of service parameters: (1) Update the number of instances or replicas to implement the scaling policy; (2) update the parameter configurations of a replica for model version iteration.

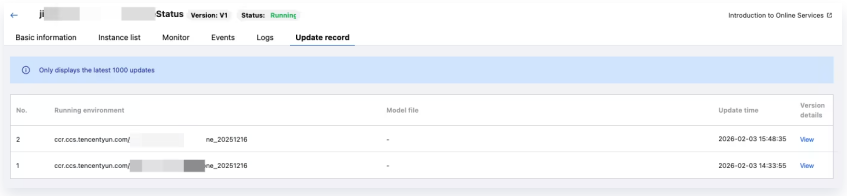
1. On the Online Services list page, click the service name to go to the version list page. Then choose Operation > **Update** of the service version to go to the service update page.

   Click **Update** to edit or modify the service parameters such as Replica Settings, Request Traffic Throttling, and Replica Scaling. Note: In the case of a single replica, updating the service will cause service unavailability during the restart process, and cannot achieve service high availability. However, in the case of multiple replicas, when the update operation is performed, the backend will perform rolling updates on these replicas in batches based on **Rolling Update Policy** specified on the parameter configuration page, which does not affect the calling of the production business for the service and can achieve high availability.





2. After confirming that the configuration information is correct, click **Update Service** to complete the update of service parameter configurations.
3. On the service version list page, click **Service Version Name** to go to the **Update History** module to view the historical update records of the current service version.

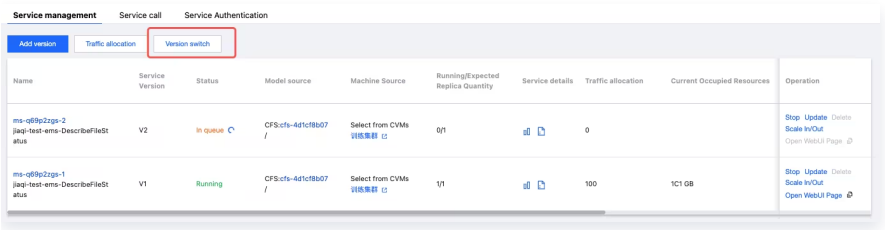


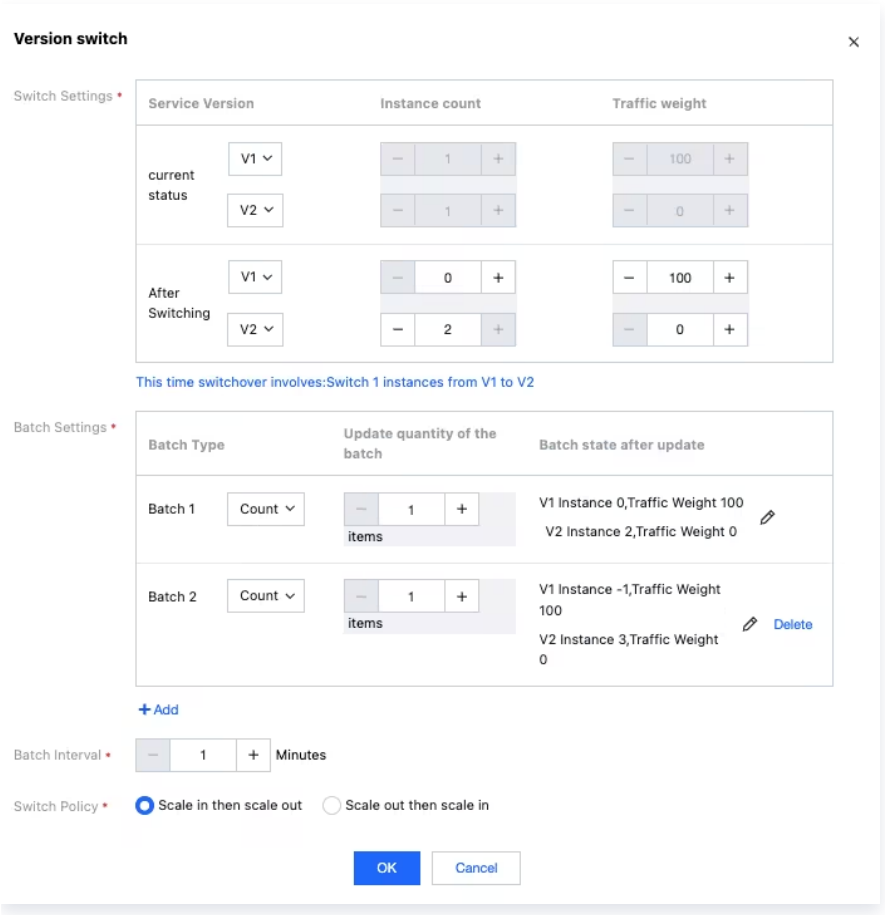
Switching Between Versions

For a service with multiple versions, click the service name to go to the version details list page. On the **Service Management** tab, you can switch between configuration versions. User scenarios are as follows:

©2013-2026 Tencent Cloud. All rights reserved.

- Assume that V1 is running and V2 is either running or not running. You can switch between configuration versions to upgrade from V1 to V2.
- Assume that V2 is running and V1 is either running or not running. You can switch between configuration versions to roll back from V2 to V1.





The core parameter configurations are described as follows:

- Switch Settings – Current Status: displays information such as Service Versions, Number of Instances, and Traffic Weight. The information is read-only and cannot be edited.
- Switch Settings – After Switching: Service Versions cannot be edited. You can adjust the number of instances and traffic weight for each version.
- Batch Settings: You can configure multiple batches to complete service switch tasks in batches. Batch Type can be set to Count or Percentage. Note: The batch type should be consistent across different batches within the same switch task.
- Batch Interval: a time interval to wait for starting the switch task for the next batch after the switch task of the previous batch is completed

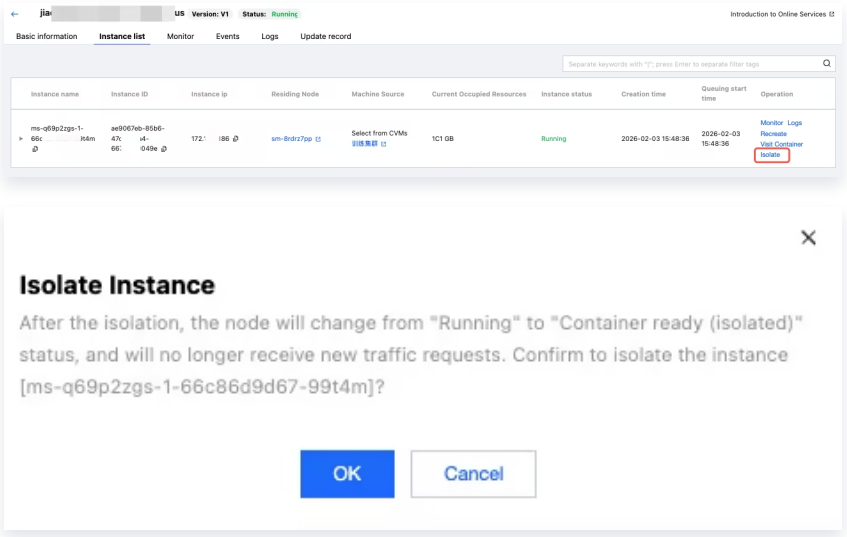
©2013-2026 Tencent Cloud. All rights reserved.

- Switch Policy:
  - Scale in then scale out: This policy is suitable for scenarios where computing resources are insufficient. You can only perform a scale-in operation to release computing resources first, and then perform a scale-out operation.
  - Scale out then scale in: This policy is suitable for scenarios with additional computing resources. You can add more computing resources first, and then perform a scale-in operation.

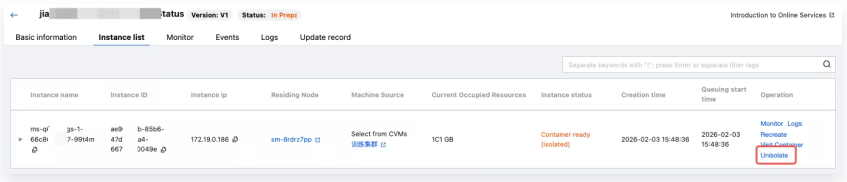
Isolating an Abnormal Instance

When an exception occurring on an instance of a service affects service quality, the platform allows you to manually isolate the traffic of this instance during diagnostics, without the need to recreate the instance Pod. This enables onsite diagnosis to be performed and allows you to manually recover traffic after the failure is fixed.

- Click the service name to go to the **Instance List** details page. For instances in the Running status, you can choose **Actions** > **Isolate** to proactively isolate a specified instance. Thereafter, traffic will no longer be routed to the isolated instance.



- For an isolated instance, you can choose **Actions** > **Unisolate** to manually restore the instance. After the isolation is successfully lifted, the instance will be restored to the Running status.



©2013-2026 Tencent Cloud. All rights reserved.

Guide on Releasing Online Services Using the Built-in Images of the Platform

Last updated: 2026-03-03 16:56:11

The platform is preset with built-in inference images compatible with mainstream deep learning and machine learning frameworks, including TensorFlow, PyTorch, and PySpark. Users can directly use its preset inference images to deploy models, which helps save the development costs for online services.

Inference Image List

Image Name	Compatible Framework	Compatible Chip
tensorflow1.15-py37	TensorFlow-1.15.0 and LIGHT-3.0	GPU (CUDA-10.0) and CPU
tensorflow2.4-py38	TensorFlow-2.4.0 and LIGHT-3.0	GPU (CUDA-11.0) and CPU
pytorch1.9.0-py38	PyTorch-1.9.0 and LIGHT-3.0	GPU(CUDA-11.1) and CPU
pytorch1.12.1-py38	PyTorch-1.12.1 and LIGHT-3.0	GPU(CUDA-11.1) and CPU
detectron2-py38	PyTorch-1.9.0 and LIGHT-3.0	GPU (CUDA-11.1)
mmdetection1.4.8-py38	PyTorch-1.9.0 and LIGHT-3.0	GPU (CUDA-11.1)
onnx1.11.1-py38	onnx-runtime-1.11.1	GPU(CUDA-11.1) and CPU
jpmml-py38	PySpark-2.4.5	CPU

Operation Guide

1. Preparing a Development Environment

a. Installing Anaconda and Preparing a Python Environment

©2013-2026 Tencent Cloud. All rights reserved.

Since the inference frameworks of Tencent Cloud TI-ONE Platform (TI-ONE) only support **Linux kernel-based systems**, you need to continue subsequent operations in a Linux kernel-based system.
To use a GPU, install CUDA locally. See Table 1 Inference Image List for the corresponding version.
Before starting the development of a model package, install Anaconda and set up a Python virtual environment. Anaconda Download Address.
After installing Anaconda, use Anaconda to create a Python virtual environment. Assume that the name of the virtual environment created this time is TIONE-test. Specify the Python version as 3.7 or 3.8 as needed. Currently, you can only install environments in these two versions.

```
conda create -n TIONE-test python=3.8
conda activate TIONE-test
```

b. Installing Dependencies Related to Inference Frameworks

TI-ONE provides an installation script for inference frameworks and related dependencies: Downloading the Script.
The main features of the script include installing the tiinfer inference framework, installing basic Python function libraries, installing environment libraries including TensorFlow and Torch, and downloading platform demos and model files. You can run `python init_infer_env.py -h` to view the method of use. For the meanings of specific parameters, see the table below. For script dependencies and existing Python environments, install CUDA on the local dev machine.

Parameter Name/Abbreviation	Description	Optional Parameter	Required
--ti-infer-version/-v	tiinfer version.	1.0	No
--model-format/-m	Model format.	TorchScript, Detectron2, ONNX, FrozenGraph, SavedModel, MMDetection, PMML, and HuggingFace	Yes
--framework/-f	Inference framework. The Torch or TensorFlow version corresponds to the model format.	torch1.9.0, torch1.12.1, onnx1.11.0, tf1.15.0, tf2.4.0, and jpmml0.6.2	Yes
--model-scene/-s	Model use case, which is detect by default.	detect, classify, nlp, ocr, and recommend	No

©2013-2026 Tencent Cloud. All rights reserved.

--demo-dir/-d	Download path. No download is performed by default.	-	No
--index-url/-i	Image source address.	-	No

Details of the environment for script installation and corresponding installation commands are shown in the following table. The content in `{}` is optional parameters. Select one parameter and enter it in the brackets.
Do not install different image environments in the same non-isolated Python environment, as this may affect the association between dependency packages. To use multiple environments, it is recommended to isolate multiple Python environments for different image names before installing them separately.

Model Format	Corresponding Image Name	Installation Dependency	Installation Command

PYTORCH TorchScript	pytorch1.9.0-py38	pytorch==1.9.0
		opencv-contrib-python==4.6.0.66
		opencv-python==4.6.0.66
		setuptools==59.5.0
		mmcv-full==1.4.8
		transformers==4.19.4
		torchvision==0.10.0
		easyocr==1.6.2
			`python init_infer_env.py --framework torch1.9.0 --model-format TorchScript --model-scene {detect, classify, ocr, nlp} --demo-dir ./`
			or
			`python init_infer_env.py --framework torch1.9.0 --model-format HuggingFace --model-scene nlp --demo-dir ./`

	pytorch1.12.1-py38	pytorch==1.12.1 opencv-contrib-python==4.6.0.66 opencv-python==4.6.0.66 mmcv-full==1.4.8 transformers==4.23.0 torchvision==0.13.1 easyocr==1.6.2	`python init_infer_env.py --framework torch1.12.1 --model-format TorchScript --model-scene {detect, classify, ocr, nlp} --demo-dir ./`
DETECTRON 2	detectron2-py38	pytorch==1.9.0 opencv-contrib-python==4.6.0.66	`python init_infer_env.py --framework torch1.9.0 --model-format Detectron2 -`

©2013-2026 Tencent Cloud. All rights reserved.

		setuptools==59.5.0 detectron2 torchvision==0.10.0	`-model-scene detect --demo-dir ./`
MMDETECTION	mmdetection1.4.8-py38	pytorch==1.9.0 opencv-contrib-python==4.6.0.66 opencv-python==4.6.0.66 setuptools==59.5.0 mmcv-full==1.4.8 transformers==4.19.4	`python init_infer_env.py --framework torch1.9.0 --model-format MMDetection --model-scene detect --demo-dir ./`
ONNX	onnx1.11.0-py38	torch==1.9.0 onnx==1.11.1 onnxruntime-gpu==1.11.1 opencv-contrib-python==4.6.0.66 setuptools==59.5.0	`python init_infer_env.py --framework onnx1.11.0 --model-format ONNX --model-scene detect --demo-dir ./`

TENSORFLOW	tensorflow2.4-py38	ensorflow==2.4.0
		opencv-contrib-python==4.6.0.66
		setuptools==59.5.0
			`python init_infer_env.py --framework tf2.4.0 --model-format FrozenGraph --model-scene nlp --demo-dir ./`
			or
			`python init_infer_env.py --framework tf2.4.0 --model-format SavedModel --model-scene {nlp, recommend} --demo-dir ./`

	tensorflow1.15-py37	tensorflow==1.15.0 opencv-contrib-python==4.6.0.66 setuptools==59.5.0	`python init_infer_env.py --framework tf1.15.0 --model-format FrozenGraph --model-scene nlp --demo-dir ./`
PMML	jpmml-py38	jpmml_evaluator==0.6.2 This environment depends on openjdk 1.8, which	`python init_infer_env.py --framework jpmml0.6.2 --model-format PMML`

©2013-2026 Tencent Cloud. All rights reserved.

		requires users to install it locally.	

2. Developing and Debugging Models Locally

Users need to develop the model_service.py file for the inference service of the platform. For details of the development method, see Model Inference Files and Their Examples. Take the TorchScript model format as an example. The standard for the model package directory and the development method of model_service.py are as follows.

- model_service.py needs to implement the functions corresponding to the tiinfer.Model class, including init (initialization), load (model loading), preprocess (data preprocessing), predict (model inference), and postprocess (postprocessing) functions.
- The init function is mainly used to set the script and model parameters.
- The load function is used to load the model and set class information. The class information is mainly used to provide data for output. You can choose whether to set class information based on your needs.
- The preprocess function reads the image used for detection from the input request and standardizes it to the input size specified by the model.
- The predict function performs the inference process.
- The postprocess function processes and packages the inference results.

PyTorch Implementation Example of a Detection Model Inference Script:

```
import os
import logging
import torch
from typing import Dict, List, Union
import numpy as np
import cv2
cv2.setNumThreads(5)
import tiinfer
import tiinfer.utils
import base64
import urllib

from utils.general import non_max_suppression, scale_coords
from torchvision import transforms

# If acceleration is used, add a reference to the acceleration library:
tiacc_inference.
# import tiacc_inference
```

©2013-2026 Tencent Cloud. All rights reserved.

```
# Model path when acceleration is not used.
PYTORCH_FILE = "model/yolov5s_ts.pt"
# Model path after acceleration.
# TIACC_FILE = "model/tiacc.pt"


class YoloV5DetectModel(tiinfer.Model):
    def __init__(self, model_dir: str):
        super().__init__(model_dir)
        # workers: platform parameter, used to set the number of
processes. The parameter is optional, and the default value is 1.
        self.workers = 1
        # The following parameters can be customized and defined based on
model needs.
        self.model = None
        self.imgsz=640  # Input image size.
        self.conf_thres=0.05
        self.iou_thres=0.45  # for NMS
        self.max_det = 20
        self.use_gpu = torch.cuda.is_available()
        self.device = torch.device('cuda:0' if self.use_gpu else 'cpu')
        self.resize = transforms.Resize((self.imgsz, self.imgsz))
        self.normalize = transforms.Normalize(mean=[0.485, 0.456, 0.406],
std=[0.229, 0.224, 0.225])

    # Load the model.
    def load(self) -> bool:
        try:
            # Assemble the model path. self.model_dir: current working
directory. Replace PYTORCH_FILE with TIACC_FILE for an acceleration model.
            model_file = os.path.join(self.model_dir,  PYTORCH_FILE)
            logging.info("model_file %s", model_file)
            # Load the model.
            self.model = torch.jit.load(model_file,
map_location=self.device)
            # Set class information.
            self.det_obj_names = ['person', 'bicycle', 'car',
'motorcycle', 'airplane', 'bus', 'train',
            'truck', 'boat', 'traffic light', 'fire hydrant', 'stop sign',
'parking meter', 'bench',
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            'bird', 'cat', 'dog', 'horse', 'sheep', 'cow', 'elephant',
'bear', 'zebra', 'giraffe',
            'backpack', 'umbrella', 'handbag', 'tie', 'suitcase',
'frisbee', 'skis', 'snowboard',
            'sports ball', 'kite', 'baseball bat', 'baseball glove',
'skateboard', 'surfboard',
            'tennis racket', 'bottle', 'wine glass', 'cup', 'fork',
'knife', 'spoon', 'bowl',
            'banana', 'apple', 'sandwich', 'orange', 'broccoli', 'carrot',
'hot dog', 'pizza', 'donut',
            'cake', 'chair', 'couch', 'potted plant', 'bed', 'dining
table', 'toilet', 'tv', 'laptop',
            'mouse', 'remote', 'keyboard', 'cell phone', 'microwave',
'oven', 'toaster', 'sink',
            'refrigerator', 'book', 'clock', 'vase', 'scissors', 'teddy
bear', 'hair drier', 'toothbrush']
            logging.info("load model to device %s" % (self.device) )
        except Exception as e:  # pylint: disable=broad-except
            logging.error("load model failed: %s", e)
            return False
        return True

    # Preprocess the request body.
    def preprocess(self, request: Dict) -> Dict:
        try:
            img_data = request["image"]
            # image_to_cv2_mat can correctly handle local images, image
URLs, and base64-encoded image content, and convert them into matrices in
OpenCV 2.
            img_mat = tiinfer.utils.image_to_cv2_mat(img_data)
            # Latitude conversion.
            img_mat = torch.Tensor(img_mat).permute((2, 0,
1)).to(self.device)

            # The image size should be within the tensor range of the
acceleration configurations in an acceleration scenario. The platform
provides the tiinfer.scale_fill_image API to proportionally scale and pad
images for user convenience.
            # For example, the tensor range set for an acceleration model
is [720, 2080], and the range for proportionally scaling images is [1080,
1999]. If the minimum side after proportional scaling is not within the
```

©2013-2026 Tencent Cloud. All rights reserved.

```
tensor range, side padding is required. For example, the minimum side
below needs to be padded to 800.
            # min_size=1080; #Minimum range of the side value for image
scaling.
            # max_size=1999; #Maximum range of the side value for image
scaling.
            # padding_limit=800; #Upper limit of the short side during
padding.
            # Guide for specific use of the image, self.scaleratio =
tiinfer.scale_fill_image(image, min_size, max_size, padding_limit,
Horizontal.RIGHT,Vertical.LOWER).

            # Data standardization.
            c, h, w = img_mat.shape
            img_tensor =
self.normalize(self.resize(img_mat).type(torch.float32)/255)
            return {"img_tensor": img_tensor.unsqueeze(0),
"original_dims": [w, h]}
        except Exception as e:
            logging.error("Preprocess failed: %s" % (e))
            return {"error": "preprocess failed"}

    # Perform inference. The request parameter is the returned result of
the preprocess function.
    def predict(self, request: Dict) -> Dict:
        with torch.no_grad():
            try:
                # Obtain the prediction results.
                out = self.model(request["img_tensor"])
                # Optional. Executed according to actual conditions.
                # Perform non-maximum suppression on the prediction
results.
                preds, nms_t = non_max_suppression(out, self.conf_thres,
self.iou_thres, max_det=self.max_det)
                data = {
                    "pred_out": preds[0].to('cpu'),
                    "original_dims": request["original_dims"]
                }
                return data
            except Exception as e:
                logging.error("Failed to predict" % (e))
```

©2013-2026 Tencent Cloud. All rights reserved.

```
                request["predict_err"] = str(e)
                return request

    # Postprocessing. The request parameter is the returned result of the
predict method.
    def postprocess(self, request: Dict) -> Dict:
        try:
            # Optional. Postprocess the prediction results based on the
model output and actual needs.
            pred_out = request["pred_out"]
            original_dim = request["original_dims"]
            # Adjust the image shape.
            pred_bbox = scale_coords([self.imgsz, self.imgsz], pred_out[:,
:4], original_dim).round()
            # Output parameters and define them based on actual network
conditions.
            labels = []
            confs = []
            label_indx = []
            boxes = []
            for i in range(pred_out.shape[0]):
                obj = pred_out[i, :].tolist()
                boxes.append(pred_bbox[i, :].tolist())
                confs.append(obj[4])
                label_indx.append(int(obj[5]))
                labels.append(self.det_obj_names[int(obj[5])])
            image_result = {
                "det_boxes": boxes,
                "det_labels": labels,
                "det_labels_idx": label_indx,
                "det_scores": confs
            }
            return {
                "result": {
                    "det_objs": image_result
                }
            }
        except Exception as e:
            logging.error("Postprocess failed: %s" % (e))
            request.pop("image", '')
            request.pop("instances", '')
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            request.pop("predictions", '')
            request["post_process_err"] = str(e)
            return request
```

3. Testing Before Deploying Online Services on TI-ONE

After installing the above environment, users can test the model on their local dev machines. The default model package path of TI-ONE inference frameworks is `/data/model/`. Before the test, confirm that the model package has been placed under this path. If needed, you can also add `export TI_MODEL_DIR=/xxx` in the system environment variable file `/etc/profile` to customize the model package path. **Point the default model package path to the folder where model_service.py is located.**
After the model package is developed, enter the path where `model_service.py` is located. Take the default framework path as an example. Run the following code in the `/data/model/` path to start the service:

```
python3 -m tiinfer --timeout 30000
```

After the service is started successfully, it can be called via port 8501. You can use test cases to call the service locally to verify its availability.

```
curl -X POST -H "Content-Type: application/json"
http://127.0.0.1:8501/v1/models/m:predict -d @test.json
```

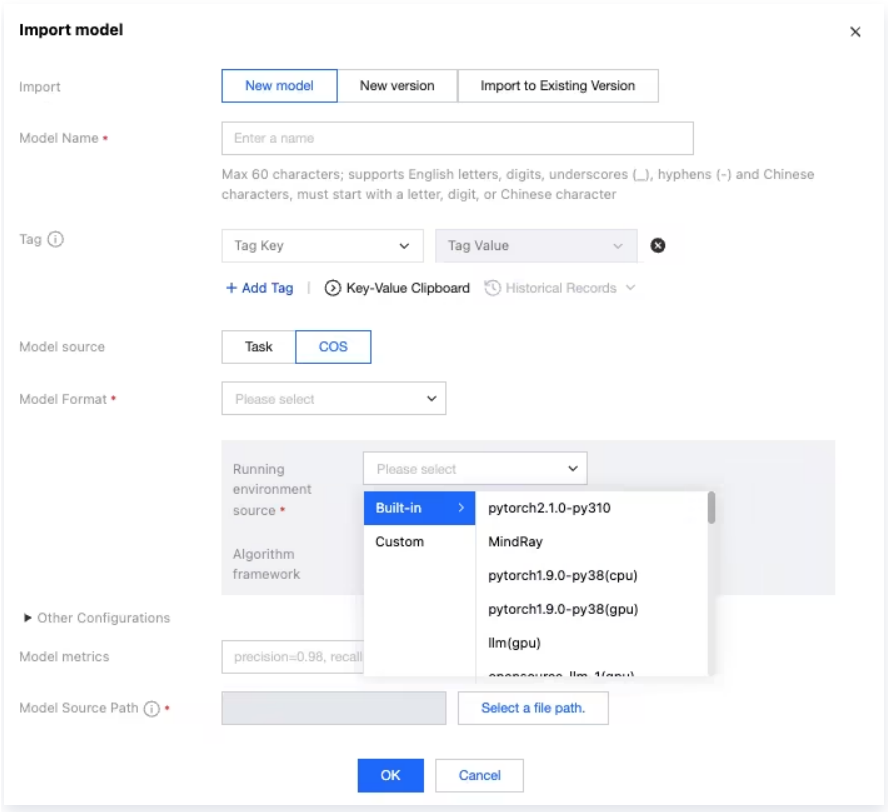
4. Deploying Model Services on TI-ONE

To use the preset inference images of the platform, you can associate the built-in running environment when importing a model in the **Model Repository** module. When the service is started with this model, the corresponding preset inference images of the platform will be automatically pulled for service deployment.

Step 1:

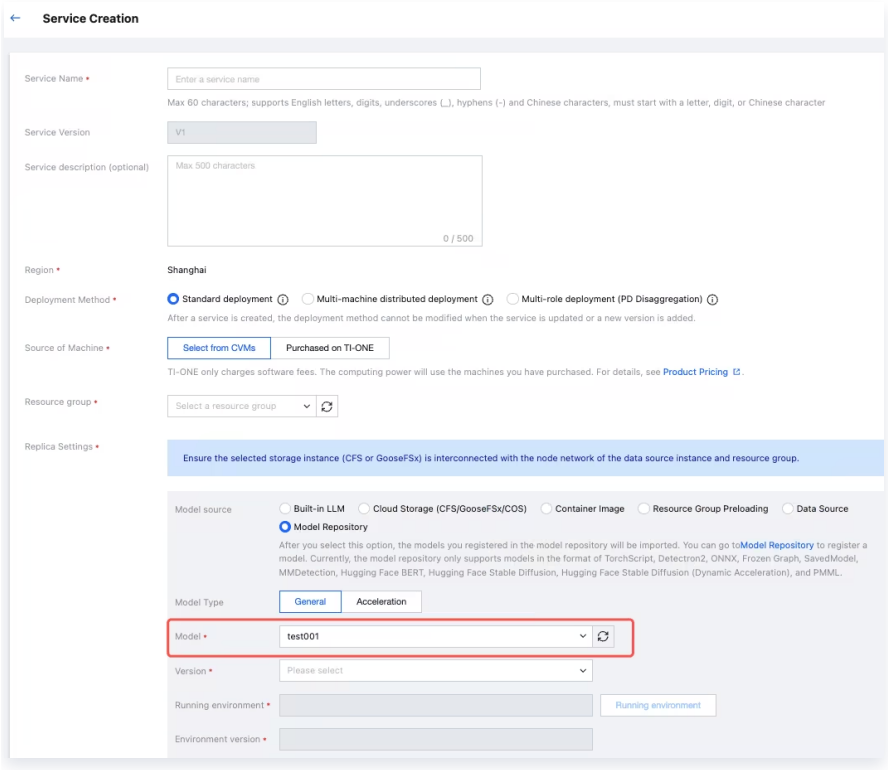
Upload the local model to Cloud Object Storage (COS) and then import it into the Model Repository module of TI-ONE.

©2013-2026 Tencent Cloud. All rights reserved.



Step 2:

Select the corresponding model from the Online Services module to release the service.



©2013–2026 Tencent Cloud. All rights reserved.

©2013-2026 Tencent Cloud. All rights reserved.

Guide on Releasing Online Services Using Custom Images

Last updated: 2026-03-03 16:56:21

Overview

This document introduces 2 methods for deploying online services on Tencent Cloud TI-ONE Platform (TI-ONE) using custom images, along with the specification constraints on deployment. Through typical cases, it demonstrates how to create custom images and release online services.

Operation Steps

Creating a Custom Image Based on tiinfer

Description of Base Images with the tiinfer Framework

The platform provides a base image for inference with the built-in tiinfer framework.
`ccr.ccs.tencentyun.com/tione-public-images/ti-cloud-gpu-base-tiinfer:py38-cu111-1.0.0`
The base image is created based on CentOS and contains the following software packages:

Software Package	Version
CUDA	11.1.1
python	3.9.13
cos-python-sdk-v5	1.9.14
coscmd	1.8.6.24
numpy	1.23.1
msgpack	1.0.5
opencv-python	4.6.0.66
opencv-contrib-python	4.6.0.66
pandas	1.4.3
Pillow	9.4.0
tiinfer	0.1.1

©2013-2026 Tencent Cloud. All rights reserved.

mosec-tiinfer	0.0.6

- Startup command of the base image: /usr/local/service/ti-cloud-infer/entrypoint.sh.
- The content of entrypoint.sh is:

```
#!/bin/bash
source /etc/profile
source /root/.bashrc
export LD_LIBRARY_PATH=/usr/local/python3/lib/python3.8/site-
packages/torch/lib:/usr/local/openmpi/lib:/usr/local/nccl/lib:/usr/local
/cuda/lib64:/usr/local/python3/lib:/usr/local/python3/lib64:/usr/local/o
penmpi/lib:/usr/local/gcc/lib:/usr/local/gcc/lib64


MODEL_DIR=/data/model

echo "================== code path ${MODEL_DIR}=========="
cd ${MODEL_DIR}

if [ -f "requirements.txt" ]; then
  echo "============== install python requirements
======================"
  echo "python3 -m pip install -r requirements.txt"
  python3 -m pip install -r requirements.txt
  echo "============== install python requirements done
================="
fi

echo "====================== start serving ============================="
echo "python3 -m tiinfer"
export TI_MODEL_DIR=${MODEL_DIR}
python3 -m tiinfer --timeout 30000
```

- The startup logic is:
  1.1 Read the requirements.txt file in the directory specified by the environment variable `${MODEL_DIR}`, and use pip to install the specified dependency Python package.
  1.2 The tiinfer framework will read the files in the directory specified by the environment variable `${MODEL_DIR}`. After loading the model, it will start an HTTP service and listen to the port defined by the environment variable `${REST_PORT}`.

©2013-2026 Tencent Cloud. All rights reserved.

1.3 When the tiinfer framework is started, the model will be loaded from the model_service.py file.

Specifications for Custom Images

1. Add a reference to the base image in the Dockerfile file. Example:
   `FROM ccr.ccs.tencentyun.com/tione-public-images/ti-cloud-gpu-base-tiinfer:py38-cu111-1.0.0`
2. The custom logic is implemented mainly by modifying the model_service.py and entrypoint.sh files.
3. When Cloud File Storage (CFS), Cloud Object Storage (COS), or Data Accelerator Goose FileSystem (GooseFS) is used as the model source, the platform places the model files (including subdirectories) from the source path in the `/data/model` directory of the service instance by default. Therefore, custom code and data must not be placed in the /data/model directory. Otherwise, they will be overwritten by the platform.

Creating an Image

This case introduces a simple adder implemented based on a base image with the tiinfer framework by modifying the model_service.py and entrypoint.sh files.
Note: In this case, the managed model repository feature provided by the platform is not used to manage the model. Instead, the model and inference code are directly packaged into the image. Therefore, it is necessary to avoid placing the model and code in the `/data/model` directory.

Writing Code

Three files are involved in total:

File	Feature
model_service.py	Writes an adder model as required by tiinfer.
entrypoint.sh	Starts scripts and allows installing more dependency packages.
Dockerfile	Copies the first two files to the image.

1. Content of model_service.py:

```
from typing import Dict
import tiinfer

class AdderModel(tiinfer.Model):
 def __init__(self, model_dir: str):
     super().__init__(model_dir)
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    def load(self) -> bool:
        self.ready = True
        return self.ready

    def preprocess(self, request: Dict) -> Dict:
        return request

    def predict(self, request: Dict) -> Dict:
        return {'result': request['a'] + request['b']}

    def postprocess(self, result: Dict) -> Dict:
        return result
```

2. Content of entrypoint.sh:

```
#!/bin/bash
source /etc/profile
source /root/.bashrc
export LD_LIBRARY_PATH=/usr/local/python3/lib/python3.8/site-
packages/torch/lib:/usr/local/openmpi/lib:/usr/local/nccl/lib:/usr/local
/cuda/lib64:/usr/local/python3/lib:/usr/local/python3/lib64:/usr/local/o
penmpi/lib:/usr/local/gcc/lib:/usr/local/gcc/lib64

MODEL_DIR=/opt/model

echo "================== code path ${MODEL_DIR}=========="
cd ${MODEL_DIR}

if [ -f "requirements.txt" ]; then
  echo "============== install python requirements
======================"
  echo "python3 -m pip install -r requirements.txt"
  python3 -m pip install -r requirements.txt
  echo "============== install python requirements done
================="
fi

echo "====================== start serving ============================="
echo "python3 -m tiinfer"
```

©2013-2026 Tencent Cloud. All rights reserved.

```
export TI_MODEL_DIR=${MODEL_DIR}
python3 -m tiinfer --timeout 30000
```

Note: The `MODEL_DIR=/opt/model` line of the above code changes the default `/data/model` of the startup directory to `/opt/model` to avoid being overwritten by the platform.

3. Content of Dockerfile:

```
FROM ccr.ccs.tencentyun.com/tione-public-images/ti-cloud-gpu-base-tiinfe
r:py38-cu111-1.0.0

COPY model_service.py /opt/model/model_service.py
COPY entrypoint.sh ./entrypoint.sh
RUN chmod +x ./entrypoint.sh
```

Note: The above code copies model_service.py to the `/opt/model` directory instead of the default `/data/model` directory to avoid being overwritten by the platform.

Packaging an Image

1. Overall steps:
   - Configure the local Docker environment and activate Tencent Container Registry (TCR).
   - Create a namespace and a personal image repository.
   - Compile a custom inference image and push it to the personal image repository.
   - When the model service is started, choose not to use the model file in the instance container column, select the running environment to enter the list of personal image repositories, and select the custom image environment pushed in the previous step.
   - Configure parameters and start the service.
2. Detailed description:

   Run the following command for packaging:

```
docker build . --tag
ccr.ccs.tencentyun.com/YOUR_NAMESPACE/YOUR_IMAGENAME
```

After packaging is completed, check whether the service is running normally in the local environment as follows:

- Run `docker run -d --name myinfer ccr.ccs.tencentyun.com/YOUR_NAMESPACE/YOUT_IMAGENAME` to run the service.
- Run `docker exec -it myinfer bash` to enter the container.

©2013-2026 Tencent Cloud. All rights reserved.

- Run `curl http://127.0.0.1:8501/v1/models/m:predict -d '{"a": 1, "b": 2}'` in the container to obtain a correct result: `{"result": 3}`.
- Exit the container, return to the local environment, and upload the image by running the following command: `docker push ccr.ccs.tencentyun.com/YOUR_NAMESPACE/YOUR_IMAGENAME`.

Creating a Custom Image with Other Inference Frameworks

The platform supports deploying the online services of models using custom images with other inference frameworks.

Specifications for Custom Images

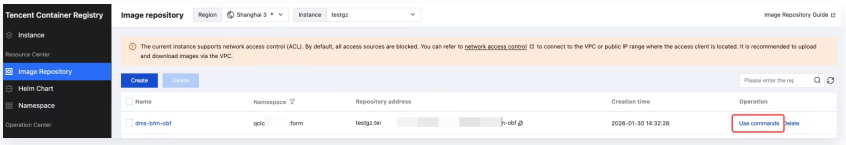
1. The service must receive requests via the HTTP protocol, and only supports POST.
2. When CFS, COS, or GooseFSx is used as the model source, the platform places the model files (including subdirectories) from the source path in the `/data/model` directory of the service instance by default. Therefore, custom code and data must not be placed in the `/data/model` directory. Otherwise, they will be overwritten by the platform.
3. The image has been verified locally and can provide services normally.

Uploading a Custom Image and Releasing an Inference Service

Uploading a Custom Image

1. Log in to TCR.
2. On the **Image Repository** page, click **Create**.
3. Upload the image.

   Click **Quick Commands** in the image repository to view the operation commands and upload the image.



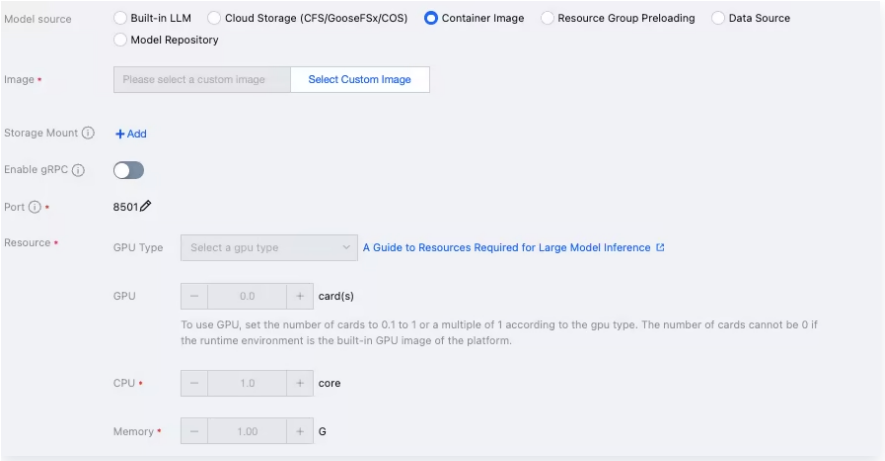
Releasing an Online Service

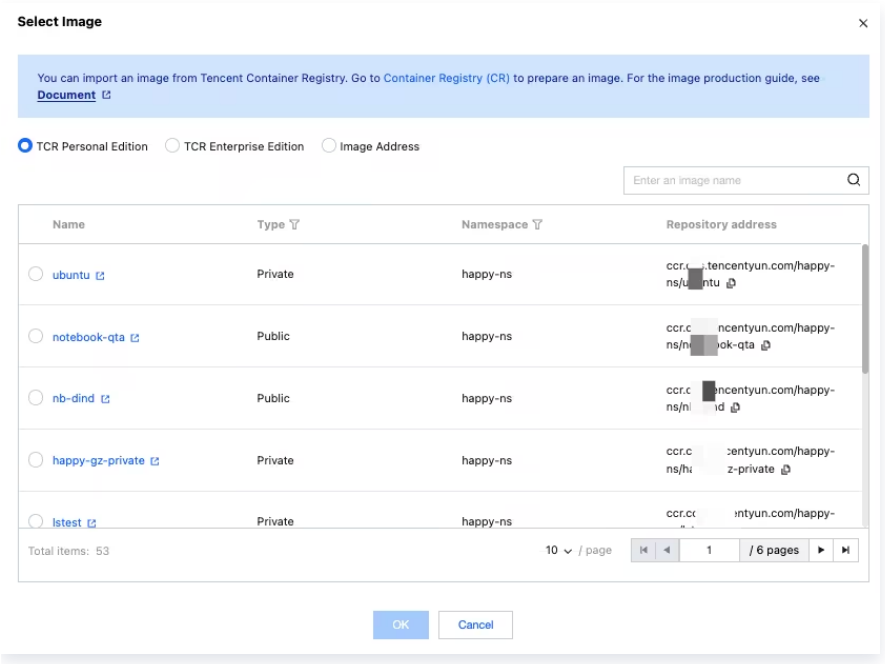
On the Online Service page of TI-ONE, **click Create Service**.

Method 1: Packaging a Model for Use in an Image

If you have packaged a model in an image, you can use the image to release a service directly. The configuration of the service replica is as follows:

©2013-2026 Tencent Cloud. All rights reserved.

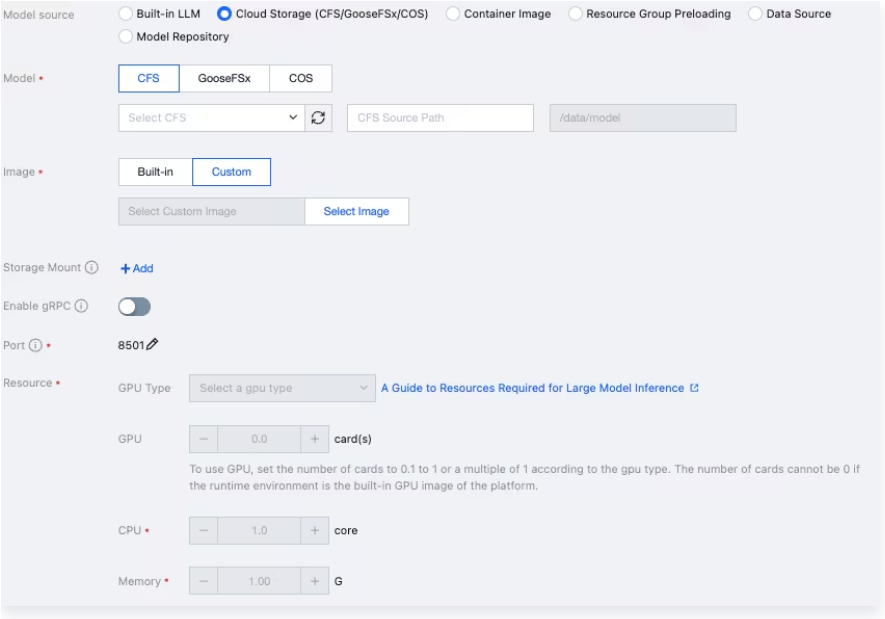




Method 2: Mounting a Model in CFS to a Container for Use (Operation Steps for GooseFSx Are Similar to Those for CFS)

If your image only serves as a running environment for a service, you can upload a model to CFS and then mount it to a container. The model will be mounted in the /data/model directory, and your service can load the model from this directory.
The configuration of the service replica is as follows:

©2013-2026 Tencent Cloud. All rights reserved.



©2013-2026 Tencent Cloud. All rights reserved.

Model Inference File Introduction and Examples

Last updated: 2026-03-03 16:56:35

Introduction to TI-ONE Built-In Inference Framework (tiinfer)

tiinfer is an inference framework used by the Model Services module of Tencent Cloud TI-ONE Platform (TI-ONE). Based on the basic capabilities provided by mosec, tiinfer offers the following features:

- High-performance HTTP services
- Flexible orchestration of the calculation process

Installation

tiinfer supports the Gnu or Linux system and depends on Python 3.8 or later versions. You need to run the following commands to install the PyPI package:

```
pip install tencent_tiinfer
pip install mosec
```

HelloWorld!

The following code shows an example of the simplest HelloWorld:

```
from typing import Dict
import tiinfer
import mosec

### Log-related code. ###
# Add the following code to print the request logs recorded by the
framework to the terminal.
import logging
import sys
logger_formatter = "%(asctime)s %(levelname)s     %(module)s:%(lineno)d
 %(message)s"
logging.basicConfig(stream=sys.stdout, format=logger_formatter,
level=logging.DEBUG)
### Log-related code. ###

# tiinfer supports the native mosec.Worker.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
class HelloWorld(mosec.Worker):
    def forward(self, req: Dict) -> Dict:
        return {"hello": f"world. raw req is {req}"}

# Launch two processes to process requests simultaneously.
tiinfer.append_worker(HelloWorld, num=2)
```

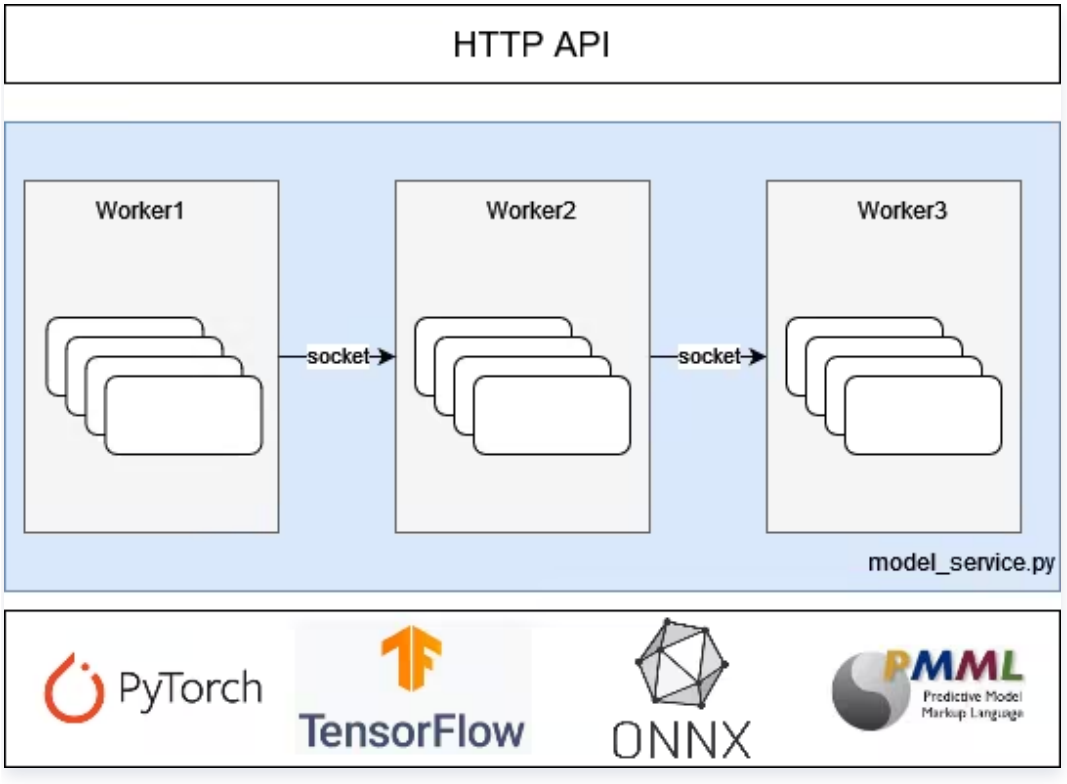
Save the preceding code as the model_service.py file, and run the following commands to start the entire inference service:

```
TI_MODEL_DIR=`pwd` python3 -m tiinfer --timeout 30000 --port 8501
```

The value of the timeout configuration item indicates the web service timeout in milliseconds. After the model is started, an HTTP service will be launched on port 8501, and the request address is `/v1/models/m:predict`. Run the following command to access:

```
> curl -X POST -d '{"key": "values"}' http://localhost:8501/inference
{
  "hello": "world. raw is {'key': 'values'}"
}
```

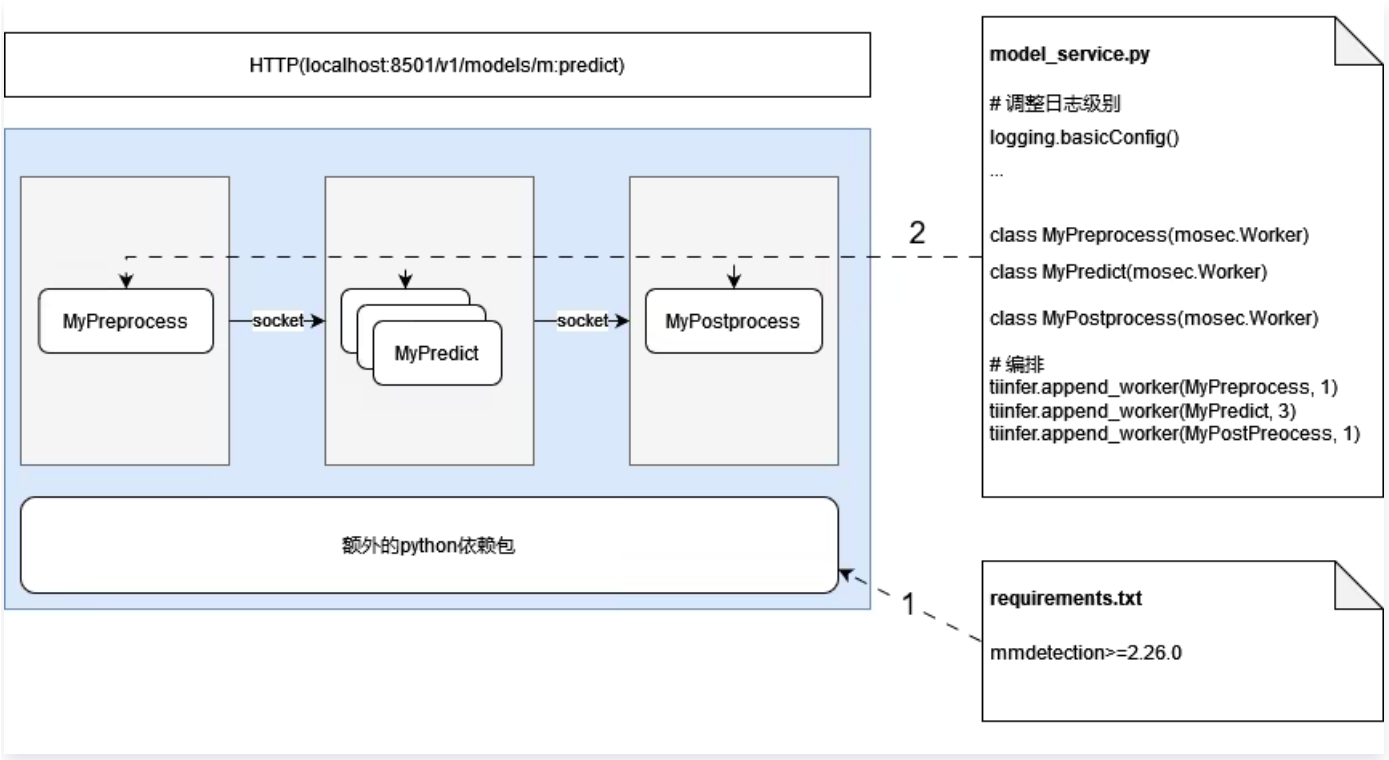
Architecture of tiinfer



©2013-2026 Tencent Cloud. All rights reserved.

To reduce the model launch difficulty, the tiinfer image provided by the platform has already packaged many popular inference engines. It will read the model_service.py file provided by the customer and automatically start an HTTP service. The specific process for starting the service is as follows:

1. Use pip to install Python dependencies specified in the requirements.txt file.
2. Read the model_service.py file and instantiate several processes for processing procedures.
3. Start the HTTP service.



Environment Variables Built in the Framework

The framework operates in single-process mode by default. You can adjust the environment variables built in the framework as needed to enable multi-process mode and make full use of resources.

Environment Variable	Description	Default Value
TI_MODEL_DIR	Model path	/data/model/
TI_PREPROCESS_NUMS	Number of preprocessing processes	0
TI_INFERENCE_NUMS	Number of inference processes	1

©2013-2026 Tencent Cloud. All rights reserved.

TI_POSTPROCESS_NUMS	Number of postprocessing processes	0
TI_INFERENCE_MAX_BATCH_SIZE	Number of inference batches	1

Note:

1. When TI_PREPROCESS_NUMS==0 and TI_POSTPROCESS_NUMS==0, the preprocess, predict, and postprocess functions in the model_service.py file are executed in a process.
2. Otherwise, the preprocess, predict, and postprocess functions in the model_service.py file are executed in different processes respectively.
3. The load and predict functions are executed in the same process.
4. Note that the value of TI_INFERENCE_MAX_BATCH_SIZE should be adjusted to avoid triggering a GPU out-of-memory error.

Customizing an Inference Worker

Each inference worker needs to inherit mosec.Worker to define the processing logic:

1. Optional: Override the `__init__` function to carry out some initialization work. You can only override it when necessary. Note that you need to first call `super.__init__()` to complete the parent class initialization.
2. Required: Customize the forward function to provide processing capabilities. Generally, data processing is completed within the forward function.

```
class Worker:
    def __init__(self)

    def forward(data)
```

After the inference worker is defined, you need to call the `tiinfer.append_worker()` function for orchestration:

```
def append_worker(
    worker: Type[mosec.Worker],
    num: int = 1,
    start_method: str = "spawn",
    env: Union[None, List[Dict[str, str]]] = None,
```

©2013-2026 Tencent Cloud. All rights reserved.

```
) -> None
"""
    worker: a processing worker that inherits mosec.Worker and implements
the forward method.
    num: number of processes for parallel computing (greater than or equal
to 1).
    start_method: a process startup method (spawn or fork).
    env: environment variables required before process startup.
"""
```

During a complete inference process, it is generally necessary to preprocess the input, enter it into the model for inference, postprocess the inference result, and then return it to the caller. The inference process is usually completed on the GPU, while preprocessing and postprocessing often involve CPU computations or even some I/O processing. If preprocessing, postprocessing, and inference are handled in the same process, the number of started processes is mainly limited by the GPU memory and computing power of the GPU. At this point, separating preprocessing and postprocessing from the inference process can make full use of the processing capability of the CPU. See the following code snippets:

```
import tiinfer
from mosec import Worker
from typing import Dict, Any


class MyPreprocess(Worker):
    def forward(self, data: Dict) -> Any:
      # The input is a Dict converted from JSON and requires some
necessary preprocessing.


class MyPredict(Worker):
    def __int__(self):
      super.__init__()
      # Read and load the model.

    def forward(self, data: Any) -> Any:
      # The input is the result of preprocessing. The inference result is
obtained after inference by calling the model.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
class MyPostprocess(Worker):
    def forward(self, data: Any) -> Dict:
      # The input is the inference result. You can convert it into a Dict
through postprocessing and return it as JSON to the caller.


# Orchestrate the processing procedures: 1 preprocessing worker > 2
inference workers > 1 postprocessing worker.
tiinfer.append_worker(MyPreprocess, 1)
tiinfer.append_worker(MyPredict, 2)
tiinfer.append_worker(MyPostprocess, 1)
```

Overview of model_service.py

Requirements for model_service.py

See the relevant chapters in the tiinfer introduction mentioned above.

PyTorch Implementation Example of a Classification Model Inference Script

```


import logging
import os
import time
from typing import Dict, List
from urllib.request import urlretrieve

import cv2  # type: ignore
import numpy as np  # type: ignore

import torch  # type: ignore

import tiinfer
import tiinfer.utils
import mosec


### Log-related code. ###
# Add the following code to print the request logs recorded by the
framework to the terminal.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
logger = logging.getLogger()
logger.setLevel(logging.DEBUG)
formatter = logging.Formatter(
    "%(asctime)s - %(process)d - %(levelname)s - %(filename)s:%(lineno)s -
%(message)s"
)
sh = logging.StreamHandler()
sh.setFormatter(formatter)
logger.addHandler(sh)
### Log-related code. ###

# The preprocessing procedure decodes the input base64 string and performs
some scaling operations based on the requirements of the model.
class Pre(mosec.Worker):
    def forward(self, req: Dict) -> cv2.Mat:
        # The input data for preprocessing is a Dict converted from JSON.
        img_base64_bytes = req["image"]
        img = tiinfer.utils.image_to_cv2_mat(img_base64_bytes)
        # bgr -> rgb
        img = img[:, :, ::-1]
        # Perform image preprocessing.
        img = cv2.resize(img, (256, 256))
        crop_img = (
            img[16 : 16 + 224, 16 : 16 + 224].astype(np.float32) / 255
        )  # center crop
        crop_img -= [0.485, 0.456, 0.406]
        crop_img /= [0.229, 0.224, 0.225]
        crop_img = np.transpose(crop_img, (2, 0, 1))
        return crop_img

# Load the model, perform inference on the preprocessed result, convert it
into the final result, and then pass the final result to the caller.
class Infer(mosec.Worker):
    def __init__(self) -> None:
        super().__init__()
        # Obtain the current directory and load the model based on the
directory.
        self.root_path = tiinfer.TI_MODEL_DIR
        # Preferentially use a GPU.
        self.device = (
```

©2013-2026 Tencent Cloud. All rights reserved.

```
            torch.device("cuda") if torch.cuda.is_available() else
torch.device("cpu")
        )
        ### Start loading non-accelerated models.###
        # The model exists in the model directory.
        model_file = os.path.join(self.root_path, "model/resNet50.pt")
        ### End loading non-accelerated models.###

        ### Start loading accelerated models.###
        # #Additionally import tiacc_inference for accelerated models.
        # import tiacc_inference
        # model_file = os.path.join(self.root_path, "model/tiacc.pt")
        ### End loading accelerated models.###

        # Load the model.
        self.model = torch.jit.load(model_file)
        self.model.eval()

        # Load the list of categories for this classification before the
model performs classification and inference.
        self.categories = load_categories()

    def forward(self, img: cv2.Mat) -> Dict:
        with torch.no_grad():
            batch = torch.stack(
                [torch.tensor(arr, device=self.device) for arr in [img]]
            )
            pred_results = self.model(batch)
            prob = torch.nn.functional.softmax(pred_results, dim=1)
            top1_prob, top1_catid = torch.topk(prob, 1)
            return [
                {
                    "confidence": top1_prob[i].tolist()[0],
                    "pred": self.categories[top1_catid[i].tolist()[0]],
                }
                for i in range(top1_prob.size(0))
            ][0]


# Read the category information corresponding to the tag ID from the tag
file.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
def load_categories() -> List[str]:
    logging.info("loading categories file...")
    local_filename = "imagenet_classes.txt"
    if not os.path.exists("imagenet_classes.txt"):
        local_filename, _ = urlretrieve(

"https://raw.githubusercontent.com/pytorch/hub/master/imagenet_classes.txt
"
        )
    with open(local_filename, encoding="utf8") as file:
        return list(map(lambda x: x.strip(), file.readlines()))

# Orchestrate the processing procedures: 2 processes for preprocessing > 1
process for inference.
tiinfer.append_worker(Pre, 2)
tiinfer.append_worker(Infer, 1)
```

Downloading Demos

The platform provides demos for various models. The following table lists the download addresses:

Format	Scenario	Download Address
TorchScript	Classification	Download
TorchScript	Detection	Download
TorchScript	NLP	Download
TorchScript	OCR	Download
Detectron2	Detection	Download
MMDetection	Detection	Download
HuggingFace	NLP	Download
SavedModel	NLP	Download
SavedModel	Recommendation	Download
FrozenGraph	NLP	Download
ONNX	Detection	Download

©2013-2026 Tencent Cloud. All rights reserved.

Introduction to the TI-ACC Inference Acceleration Function

tiacc_inference.load()

For models optimized in Detectron2 or MMDetection format on TI-ONE, you need to use the tiacc-inference.load() function to load them. For models optimized in other formats, you do not need to use the tiacc_inference function to load them, but you can use the native method to load them.

- MMDetection
  Native example:

```
from mmdet.apis import init_detector
model = init_detector(config, checkpoint, device=device)
```

  Usage example after optimization:

```
import tiacc_inference
model = tiacc_inference.load('tiacc.pt')  #tiacc.pt is the new model
generated after model optimization.
```

- Detectron2 (for PyTorch models exported by Detectron2)
- Native example:

```
import torch
model = torch.load(checkpoint) #A model file in .pth format.
```

  Usage example after optimization:

```
import tiacc_inference
model = tiacc_inference.load('tiacc.pt')  #tiacc.pt is the new model
generated after model optimization.
```

- Detectron2 (for models constructed through Detectron2.modeling.build_model)
- Native example:

```
from detectron2.config import get_cfg
from detectron2.modeling import build_model
cfg = get_cfg()
cfg.MODEL.DEVICE = device
cfg.MODEL.WEIGHTS = checkpoint
```

©2013-2026 Tencent Cloud. All rights reserved.

```
model = build_model(cfg)
```

Usage example after optimization:

```
import tiacc_inference
model = tiacc_inference.load('tiacc.pt') #tiacc.pt is the new model
generated after model optimization.
```

©2013-2026 Tencent Cloud. All rights reserved.

Batch Prediction
Introduction to Batch Prediction

Last updated: 2026-03-10 10:20:30

For business scenarios that involve irregular batch data processing, the platform provides the Batch Prediction module capability, which supports using computing resources as needed and automatically releasing inference computing resources after batch data processing is completed.

Prerequisites

- You have trained a model and imported it into the model repository, or you have uploaded your custom inference image to Tencent Container Registry (TCR).
- You have prepared the data that requires batch prediction and uploaded it to Cloud Object Storage (COS) of Tencent Cloud.
- You have created a folder in COS to store the inference result file for batch prediction.

Creating a Prediction Task

1. Log in to the Tencent Cloud TI-ONE Platform (TI-ONE) console and choose **Model Services** > **Batch Prediction** in the left sidebar to go to the batch prediction task page.
2. On the task list page, click **Create Task** to go to the task creation page.
3. On the task creation page, configure relevant parameters for the batch prediction task.

   3.1 Basic task information

Parameter	Description
Task Name	Name of the prediction task. Enter it based on the rules prompted on the page.
Task Description	Configure a description of the batch prediction task as needed.
Region	Prediction tasks under the same account are isolated by region. The value of the region field is automatically populated with the region you select on the task list page.
Billing mode	You can select the pay-as-you-go mode or the monthly subscription mode. (A) In the yearly/monthly subscription mode, you can use the inference resource group purchased from the Resource Group Management module to deploy a task. Since the computing resource fee

©2013-2026 Tencent Cloud. All rights reserved.

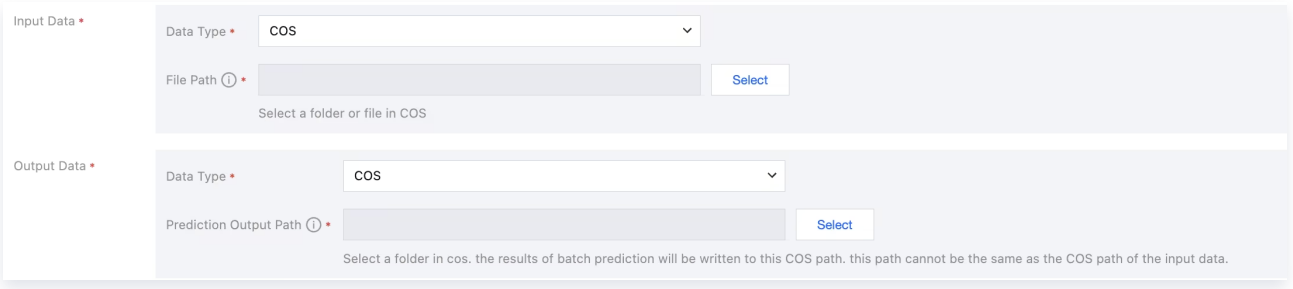
	is paid when the resource group is purchased, no fees need to be charged when the task is created. (B) In the pay-as-you-go mode, you do not need to purchase an inference resource group in advance. When a task is created, fees for the first two hours are frozen based on the Cloud Virtual Machine (CVM) instance specifications required by the task. Thereafter, fees are charged hourly based on the number of running instances.
Resource Group	If you select the yearly/monthly subscription mode, you can select an inference resource group from the Resource Group Management module.

3.2 Service instance information

Parameter	Description
Model Source	Selecting an image from the image repository applies to the scenario where the custom image required for the prediction task has encapsulated the model file, eliminating the need to mount the model file, and the custom image has been uploaded to TCR. Selecting a model from the model repository applies to the scenario where the model file required for the prediction task has been imported into the model repository. If you need to use a model file to create a task, it is recommended that the size of the model file does not exceed 16 GB.
Select Image	(A) If you select an image from the image repository, you can set the running environment to a custom image that has been uploaded to TCR. (B) If you select a model from the model repository, the running environment field will be automatically assigned a value based on the configuration of the model repository. If you need to use a custom image to create a task, it is recommended that the size of the image file does not exceed 34 GB.
Model Inference File	When selecting a model from the model repository, you need to upload a model inference file, which must be named model_service.py. The batch prediction processing logic can be customized.
Startup Command	If no startup command is specified when a batch prediction task is created, the default startup command is python3 /opt/ml/code/model_service.py.
Spec	(A) In the monthly subscription mode, you can set how many resources to apply for from the selected resource group for the current task. (B) In the pay-as-you-go mode, you can select the required CVM instance specifications for the current task as needed.

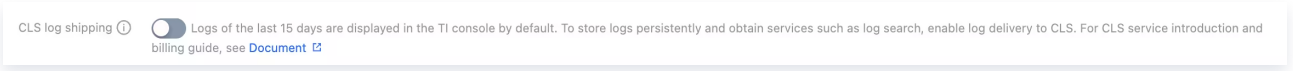
©2013-2026 Tencent Cloud. All rights reserved.

3.3 Data path configuration



Parameter	Description
Input Data Path	Storage path for the data to be processed in a batch prediction task. You can select COS. If you select COS, you need to select a folder or file in the bucket. (input data is mounted to /opt/ml/input/data by default as the prediction data directory, and your code needs to use this path to obtain the data).
Output Data Path	Target path where the batch prediction task writes the prediction result file. You can select COS . If you select COS, you need to select a folder or file in the bucket. (input data is mounted to /opt/ml/output by default as the prediction data directory, and your code needs to use this path to obtain the data).

3.4 Advanced task configuration



Parameter	Description
CLS Log Shipping	The platform provides free storage for task logs from the last 7 days. If you require persistent log storage and more flexible log search capabilities, you can enable CLS Log Shipping, and then task logs will be shipped to Cloud Log Service (CLS) of Tencent Cloud based on the logset and log topic.
Tag	You can add tags to tasks for authorization or billing based on tags.

4. After confirming that the prediction task configuration is correct, click **Save** to complete the creation of the task.
   During task startup, computing resources will be scheduled for you, which may take a period of time. Once started successfully, the task will automatically enter the **Running** status to perform batch prediction processing for you. After processing is completed, the task status will change to **Completed**.

©2013-2026 Tencent Cloud. All rights reserved.

Viewing the Prediction Result

When the batch prediction task is in the **Completed** status, you can view its running result.

1. On the batch prediction task page, click the name of the task to go to the task details page and view the output data path.
2. Go to the COS console, and view the result file of the prediction task via the output data path associated with it. You can preview or download the result file in the COS console.

©2013-2026 Tencent Cloud. All rights reserved.

Platform Management
Data Source Management

Last updated: 2026-02-26 16:04:09

Overview

Data Source Management allows you to register a specified directory from a storage instance (Cloud File Storage (CFS)/GooseFSx) as a data source, enabling quick mounting. Additionally, it supports implementing permission control for storage directories through Cloud Access Management (CAM) and tags.

Data Source

A data source refers to a specific directory of a storage instance. Specifically:

- **Storage instance**: refers to a storage instance of a cloud product, such as CFS and GooseFSx on Tencent Cloud.
- **Directory**: represents the specific path of data storage. Either the root directory or any subdirectory can be registered as a data source.

Data Source Management

It provides the capability to create and manage data sources, with details as follows:

- **Create data source**: Supports registering storage instances purchased on Tencent Cloud to the Tencent Cloud TI-ONE Platform and creating specified directories as data sources.
- **Quickly mount data source**: In Task-based Modeling, Dev Machines, Online Services, and other functional modules, supports one-click selection of specified directories during mounting, eliminating the need for repeated input.
- **Directory permission control**: Supports using CAM and tags to implement permission control for storage directories. When CFS/GooseFSx is mounted within TI-ONE, you can only access data sources with specified tags.

Prerequisites

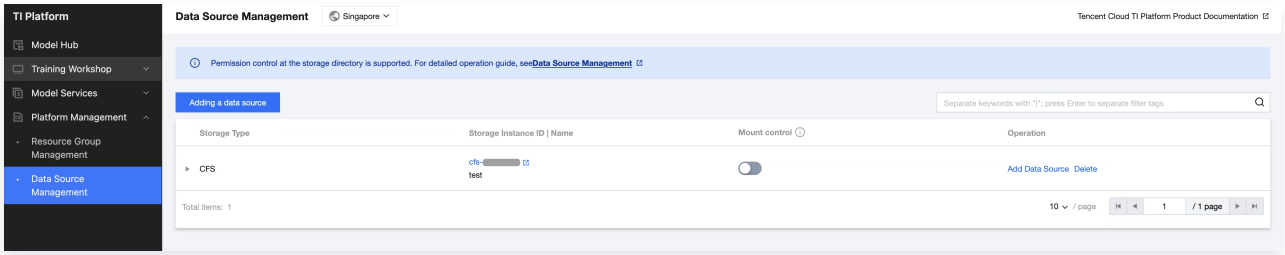
Before you use the Data Source Management feature, go to the CAM console to authorize the following CAM policy to the specified user/user group. The preset policies are as follows:

- Preset policy for read/write permissions: QcloudTIONEResouceGroupFullAccessContainMultiservice
- Preset policy for read-only permissions: QcloudTIONEReadOnlyAccessContainMultiservice

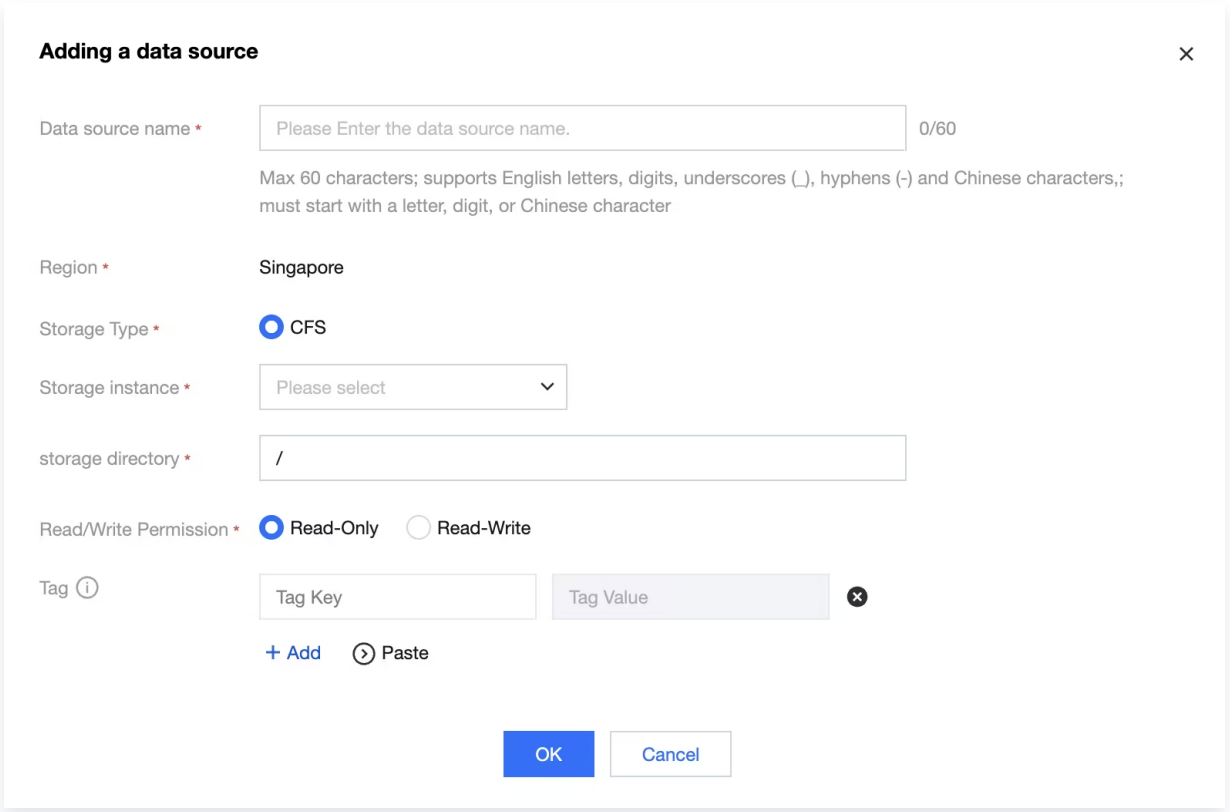
Creating a Data Source

©2013-2026 Tencent Cloud. All rights reserved.

1. Choose **Platform Management > Data Source Management** in the left sidebar to go to the **Data Source Management** page, and click **New Data Source**.



2. In the **New Data Source** pop-up window, fill in the relevant information, and click **Confirm**.



Configuration parameters and their descriptions are listed in the table below.

Field Name	Description
Data Source Name	1. Required. The length must not exceed 60 characters. Only Chinese, English, digits, underscores (_), and hyphens (-) are allowed. It must start with a Chinese character, English letter, or digit. 2. Duplicate name verification: The Data Source Name cannot be duplicated (across all storage instances). Additionally, the combination of storage directory + read/write permissions serves as a unique key, meaning that the storage directory and the read/write permissions cannot be repeated at the same time.

©2013-2026 Tencent Cloud. All rights reserved.

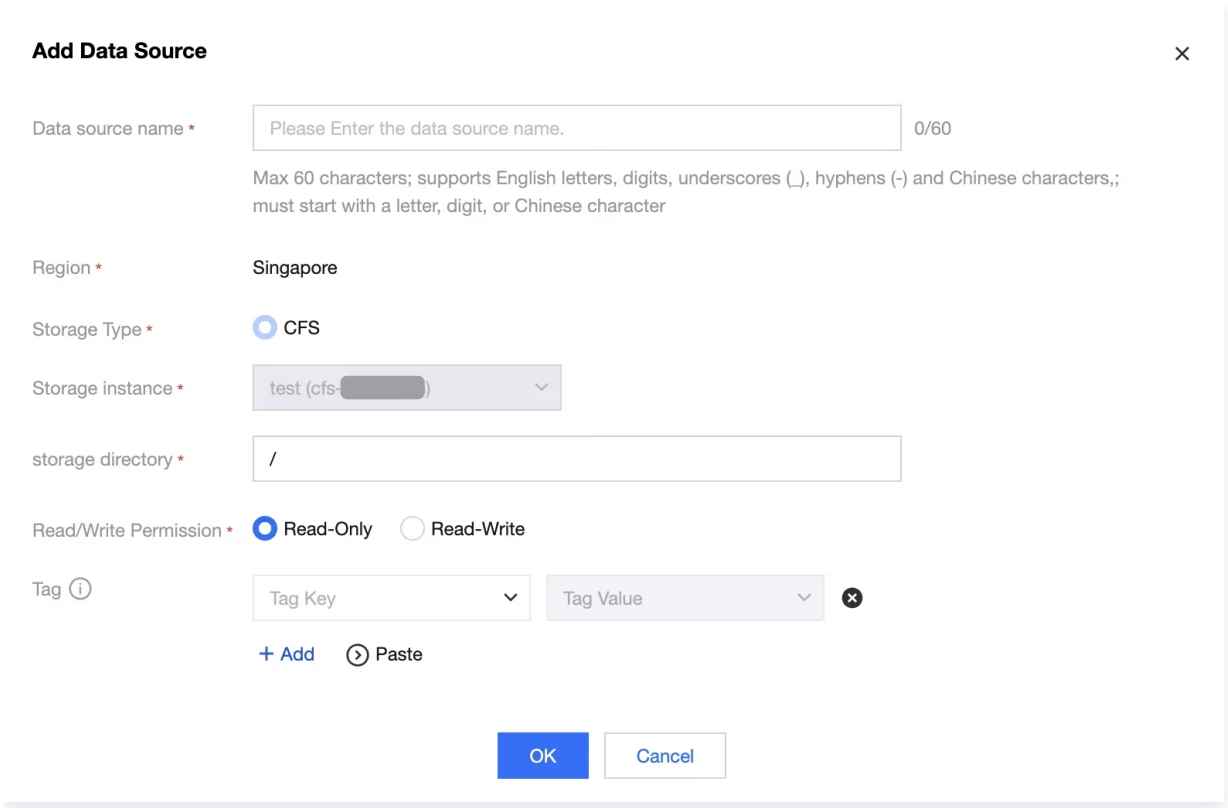
Storage Type	Required. Options include GooseFSx and CFS. CFS is selected by default.
Region	Displays the current region of the TI-ONE console, which cannot be modified.
Storage Instance	1. Required. The options are sourced from the information of purchased storage instances under the corresponding storage type. (Remarks: The text outside the parentheses indicates the name of the storage instance, and the text inside indicates the storage instance ID.) 2. **Note:** The instance here refers to the storage instance already purchased on Tencent Cloud. TI-ONE only registers and binds it, rather than creating a separate instance.
Storage Directory	Required. Supports custom file paths. The path must start with a forward slash (/) and can only contain Chinese, English, digits, underscores (_), hyphens (-), and periods (.). It cannot contain "..". (Remarks: Only one directory can be configured per data source.)
Read/Write Permissions	Required. Options include: Read-only, Read-write. Indicates the read/write permissions when modules mount the data source, including read-only mounting and read/write mounting.
Tag	1. Supports adding tags to data sources. Tag options are sourced from the tag console. Multiple tags can be added to a data source. 2. After tags are applied, you can control the visibility permissions of data sources by using tags in conjunction with CAM policies to achieve data isolation.

Managing Data Sources

Adding a Data Source

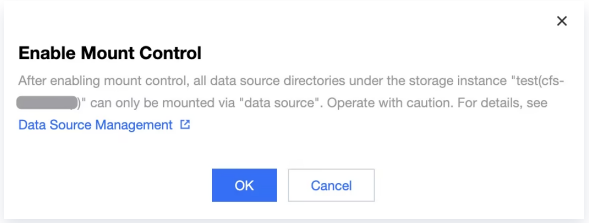
The configuration parameters are consistent with those for creating a data source. You can add a data source for a specified storage instance. The type and name/ID of the selected storage instance cannot be modified.

©2013-2026 Tencent Cloud. All rights reserved.

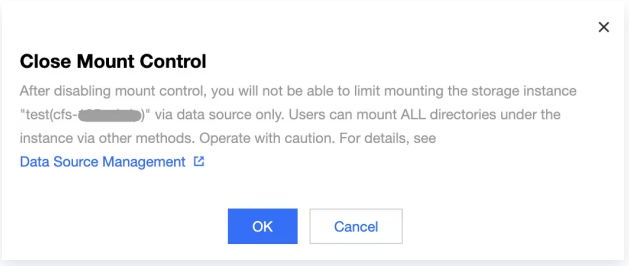


Mount Control

- Enabling mount control restricts users to accessing specified instances and storage paths only by mounting **data sources**; disabling it allows users to access any directory via the original method of specifying **storage instance + source path**.
- The effective scope of mount control is at the level of the storage instance, which means that all data sources configured under that instance will be restricted.
- **Note: This feature retains the original mounting method while providing some permission management capabilities. It is recommended to enable it in scenarios with high requirements for permission control.**



Enable mount control



Disable mount control

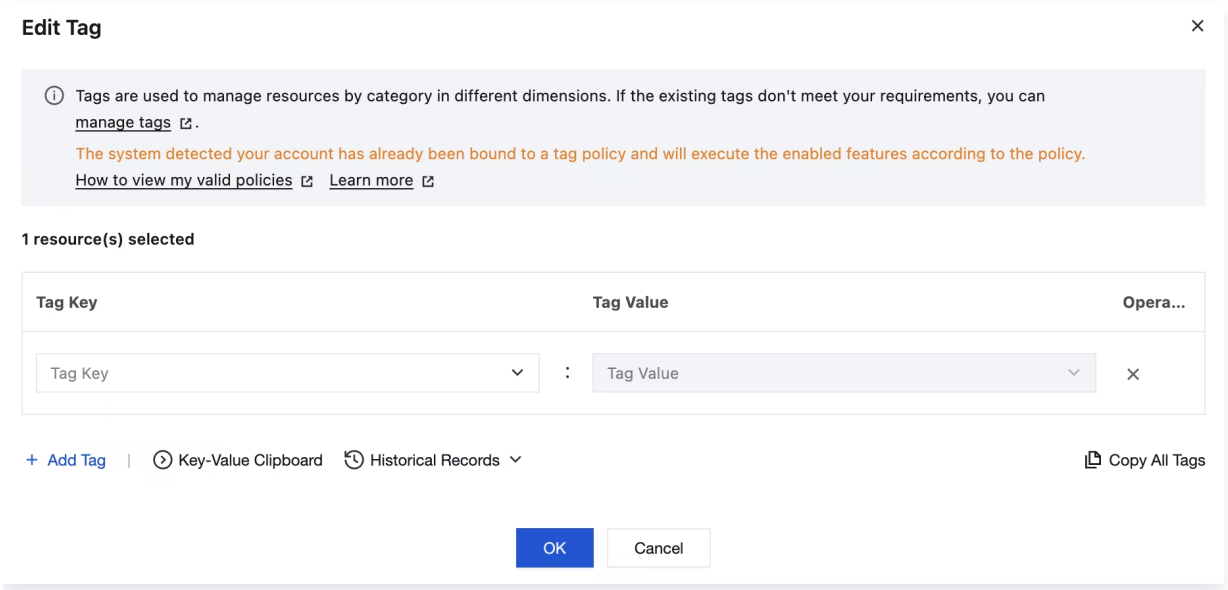
Tags & Permissions

Note:

©2013–2026 Tencent Cloud. All rights reserved.

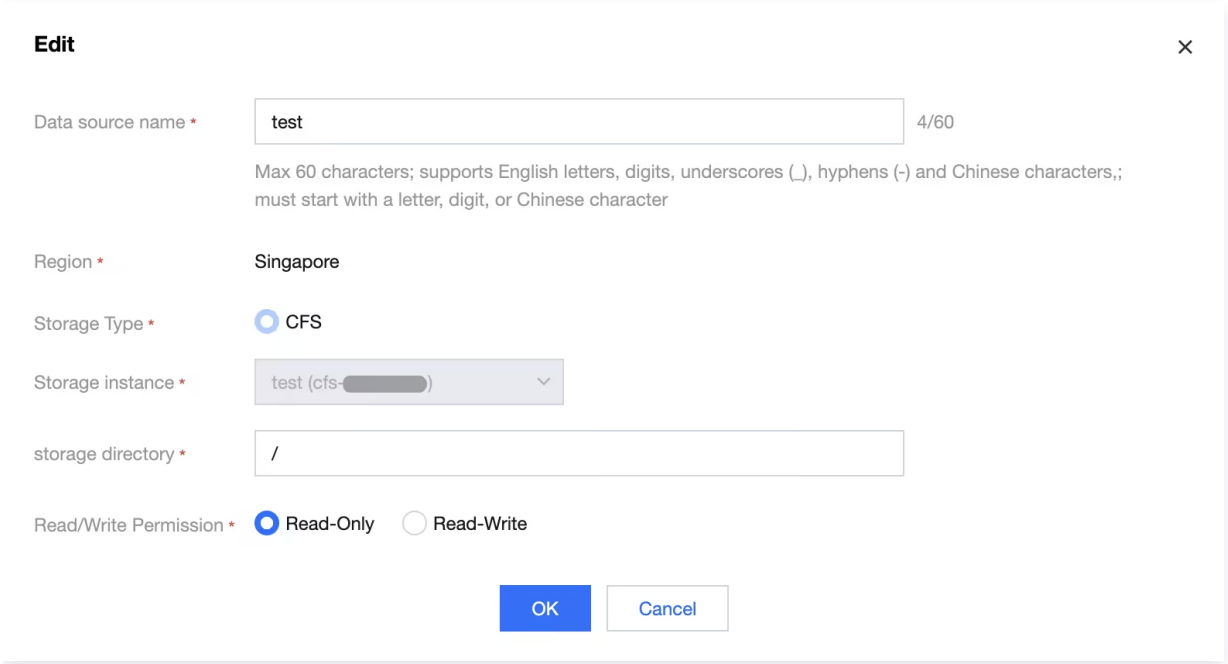
> For a tutorial on best practices for permission configuration, see Implementing Directory-Level Permission Control for Storage Using the Data Source Feature.

You can configure permissions for data sources based on CAM and tags. By assigning specific tags to data sources and configuring corresponding CAM policies, you can achieve data isolation between different Tencent Cloud sub-accounts, thereby implementing permission control for storage directories.



Editing

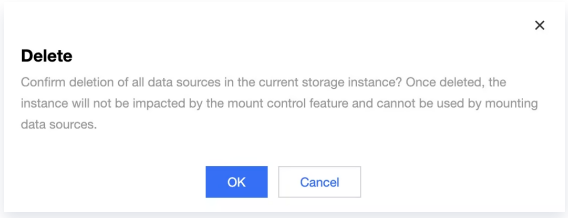
The configuration parameters are consistent with those for adding a new data source. During editing, existing information will be pre-populated. Modifications are allowed for the Data Source Name, Storage Directory, and Read/Write Permissions fields. The type and name/ID of the storage instance cannot be modified.



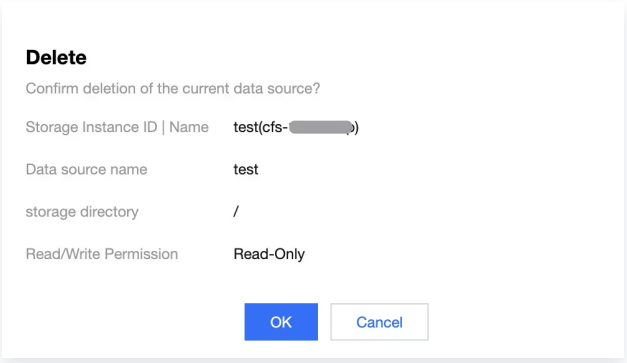
©2013-2026 Tencent Cloud. All rights reserved.

Deletion

- **Deleting a storage instance:** This means to unbind the storage instance from TI-ONE Data Source Management, not to actually delete the storage instance or its data. This action will clear all data sources configured under the storage instance.
- **Deleting a data source:** This means to delete a specific data source directory under the storage instance.



Delete a storage instance

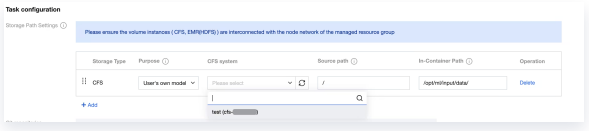


Delete a data source

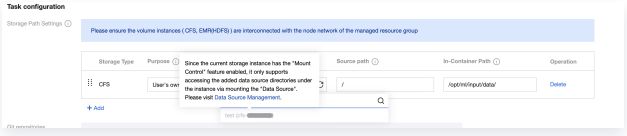
Using Data Sources

After data source configuration is completed, when storage paths are mounted in modules such as **Data Set, Task-based Modeling, Dev Machines, Online Services, and Model Evaluation**, the usage method is as follows:

1. When CFS/GooseFSx instances are mounted in the above modules using the original method, it is necessary to restrict whether users can select the corresponding instances based on whether the mount control feature is enabled for the storage instance.
   - If a restriction has been applied to an instance, the instance option will be disabled in the dropdown list, and the following message will be displayed. If no restriction is applied, the instance can be selected normally.



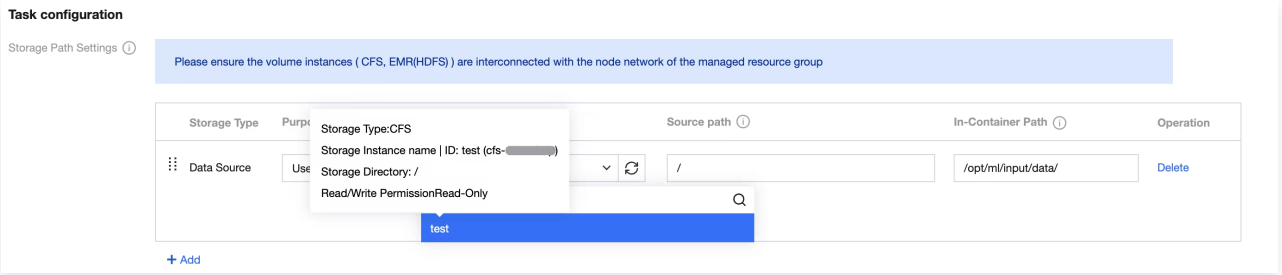
Restrict options based on whether mount control is enabled



Prompt after a restriction is applied

2. When the mount type is set to Data Source, one-click selection of specified directories is supported, eliminating the need for repeated input.
   - When you hover over a dropdown option, the storage type, storage instance name/ID, storage directory, and read/write permissions of the corresponding data will be displayed.
   - After an option is selected, the storage directory and read/write permissions of the selected data will be displayed below the text box.

©2013-2026 Tencent Cloud. All rights reserved.



3. Similarly, when data sources are mounted in business modules such as Task-based Modeling and Dev Machines, the dropdown list will only display data sources accessible to the corresponding team.

Others

1. After the launch of the Data Source Management feature, the original solution for achieving directory isolation between sub-accounts, which was solely based on the tag system, will no longer be maintained or effective.
2. Since CAM allows for complex custom policies, conflicts at the directory level might occur in extreme scenarios, as shown in the following example:
   - **Extreme scenario:** The parent directory is allowed, but the child directory is denied. Assume the directory hierarchy of the storage instance is "/", "/a", "/b", with the storage directory for Data Source A configured as "/" and Data Source B as "/a", both authorized to user Tom. Suppose a separate CAM policy is configured outside the platform to restrict Tom from accessing the storage directory "/b" of Data Source C.
   - **Platform processing logic:** Both parent and child directories are allowed. In this case, Tom can mount the directory "/b" within the platform via either the data source or the method of storage instance + source path, and can access it normally without triggering the denial of the child directory. (This is because Tom has access permission to the parent directory "/".)

©2013-2026 Tencent Cloud. All rights reserved.

Enterprise Management
Resource Group Management
Resource Group Introduction

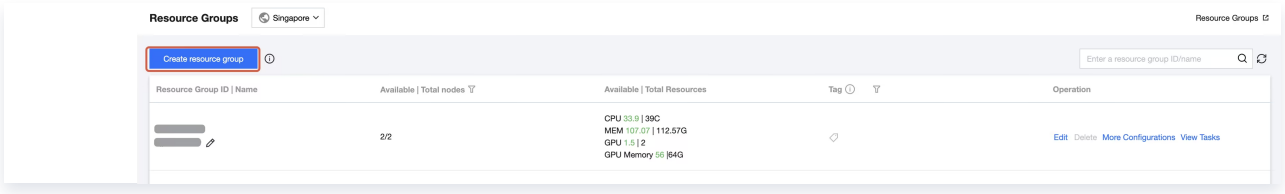
Last updated: 2026-03-12 10:38:14

Overview

Tencent Cloud TI-ONE Platform (TI-ONE) manages user-purchased Cloud Virtual Machine (CVM) instances and schedules resources in the form of resource groups. The CVM instance sources include Purchase from the TI-ONE platform and Select from CVM instances. For the differences between these two sources, please refer to Billing Overview. For the former, you can no longer add resource groups, and only existing resource groups are available. This document provides operation instructions for resource groups whose instance source is Select from CVM instances.
You can add CVM instances in the same region to the same resource group for management and then use the node resources in this resource group to start training and inference tasks.

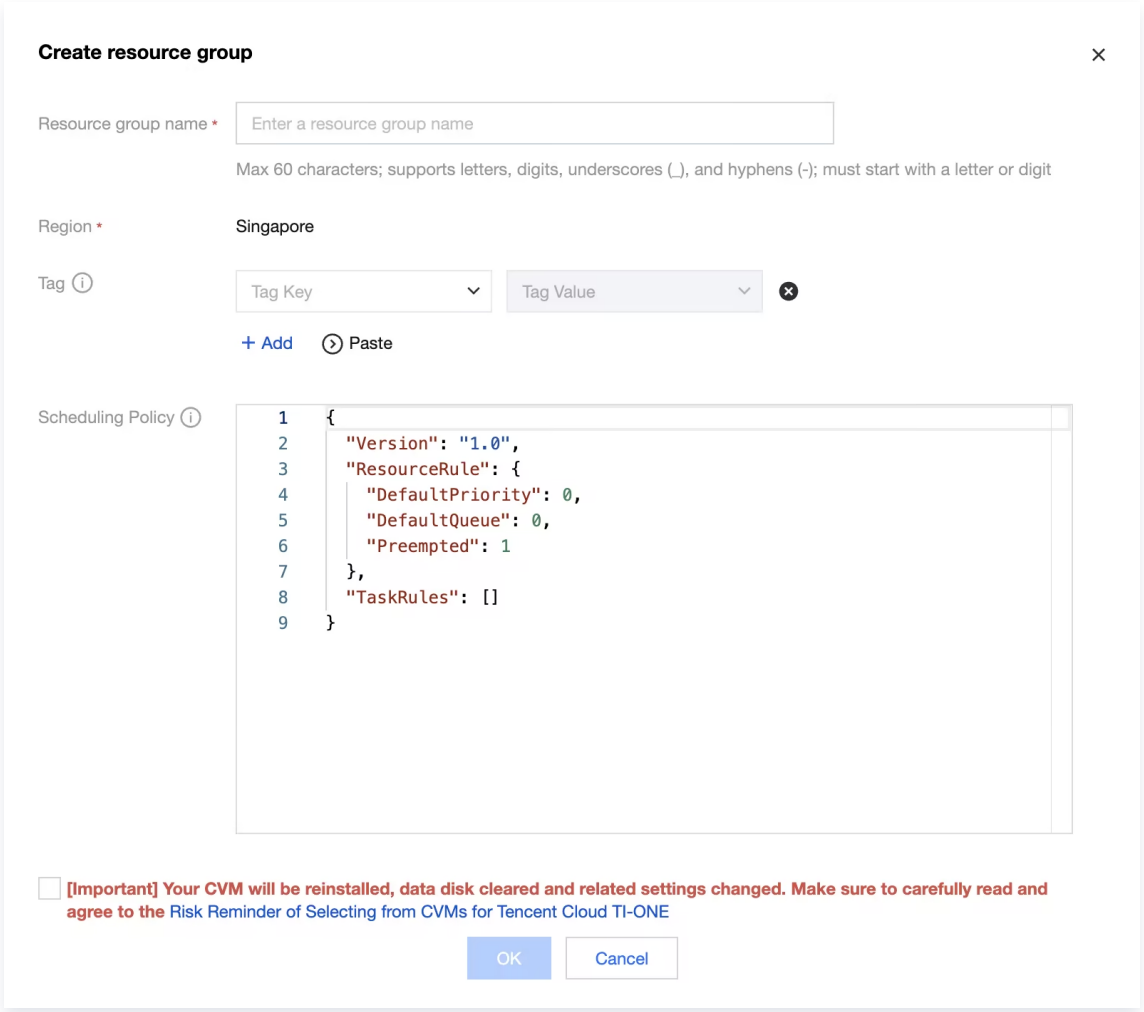
Creating a Resource Group

1. Create a Tencent Cloud account and complete identity verification.
2. Log in to the TI-ONE console with your Tencent Cloud account and go to the resource group management page.
3. Select the **region** where your business is located at the top of the page.
4. Click **Create Resource Group** to display the pop-up window for creating a resource group.



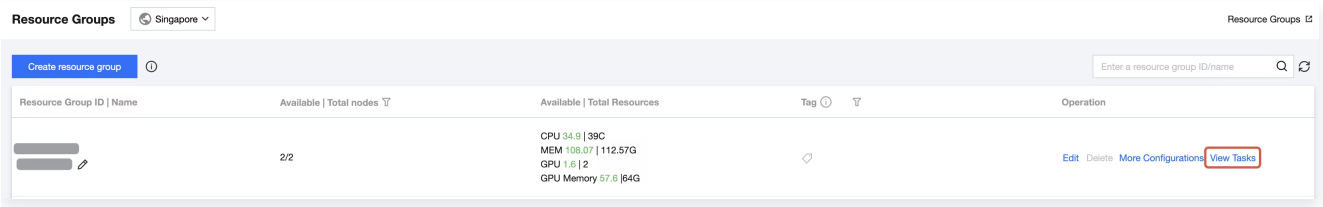
5. Enter a resource group name, check the risk warning, and click **Confirm** to complete the creation.

©2013-2026 Tencent Cloud. All rights reserved.



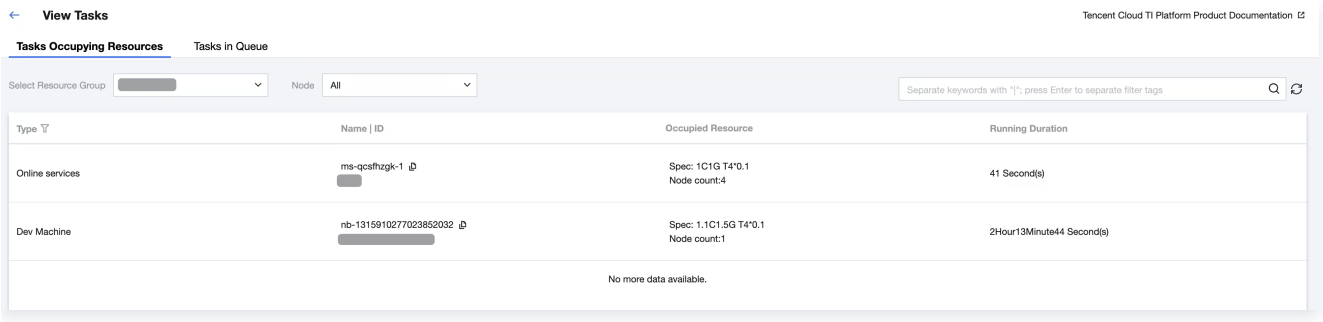
Viewing the Task List

Click **View Task List** on the resource group list page to view the tasks occupying resources and the tasks in the queue under this resource group.



Go to the **Occupied Resource Tasks** tab to view all tasks occupying resources under this resource group. You can also switch between resource groups on this page and view a specific node.

©2013-2026 Tencent Cloud. All rights reserved.

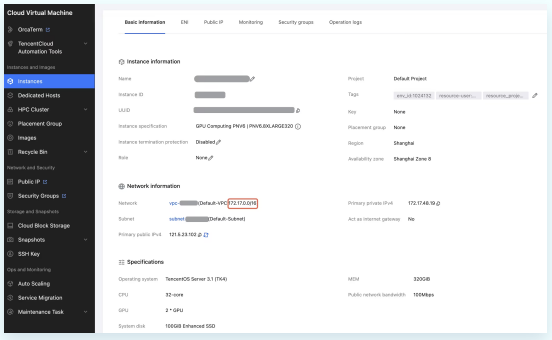


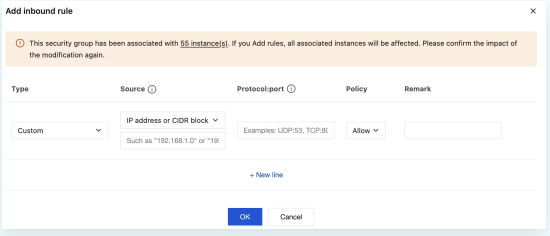
Go to the **Tasks in Queue** tab to view the training tasks queued under this resource group. Note that when the resource group's queuing policy is strict First-In-First-Out (FIFO), this sequence number indicates the order in which tasks are scheduled. When the queuing policy is non-strict FIFO, this number does not represent the actual scheduling order. For detailed rules, see Training Task Scheduling Rules.

Adding a Node

1. You can see Instance Creation Guide and go to the CVM console to purchase instances.
   - Regions supported by TI-ONE: Beijing, Shanghai, Guangzhou, and Nanjing. Purchase your instances in the regions supported by TI-ONE.
   - For the instance types supported by TI-ONE and their corresponding software subscription fees, see Billing Overview.

> **Note:**
> Important: After purchasing the CVM instance, add the Virtual Private Cloud (VPC) IP range to which the CVM instance belongs and 9.0.0.0/8 to the security group, and configure inbound and outbound rules to ensure normal communication during training/inference. The specific operation guide is as follows:
> - Purchase a CVM instance and obtain the Network information on the instance details page.
> - On the Security Group Details - Security Group Rules tab page, add two inbound and two outbound rules. Set Source to the VPC IP range in the Network field of the previous step and 9.0.0.0/8, and set Protocol Port to ALL.

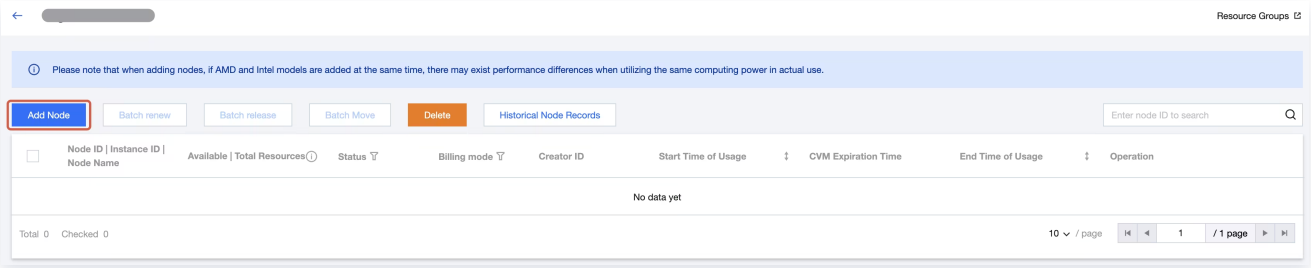




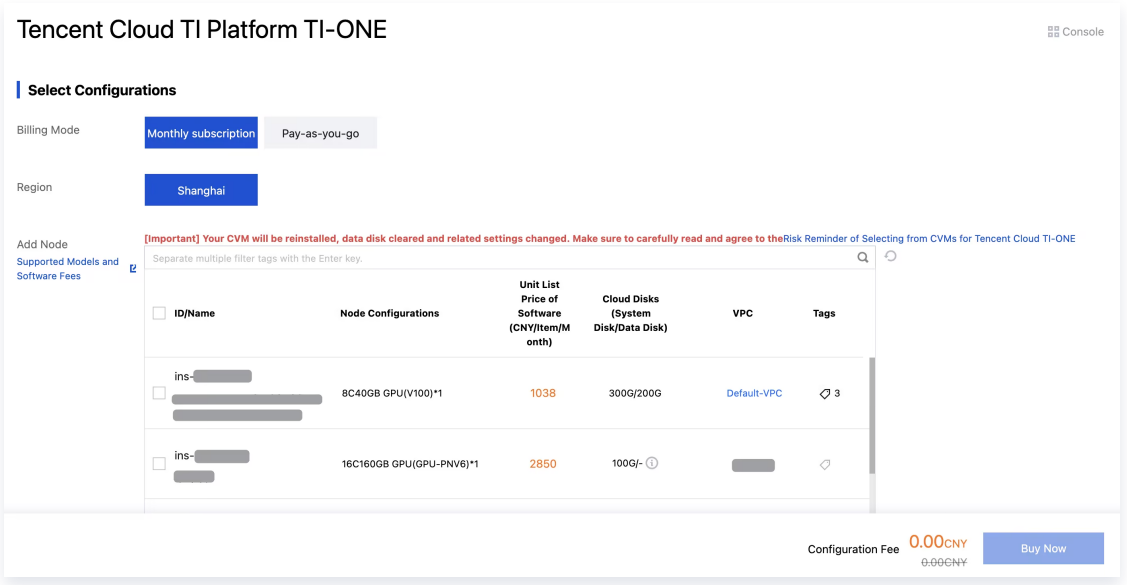
©2013-2026 Tencent Cloud. All rights reserved.

2. Click the name of a resource group whose instance source is Select from CVM instances, and then click **Add Node** to go to the purchase page.

> **Note:**
> The CVM instance resources you purchased from the CVM console will all be added to a resource group as nodes for unified management.
> If a node in a resource group is released, the resource group still exists, and you can add other nodes to it. Nodes released from resource group A can also be re-added to resource group B.
> We also provide a node migration feature, which means you can move nodes from resource group A to resource group B.



3. On the purchase page, configure the following information as prompted.



Field	Description
Billing Mode	Yearly/Monthly Subscription or Pay-as-You-Go
Region	This field displays the region where this resource group resides by default.
Nodes to Add	You can select the CVM instances to be used in TI-ONE from the instances available in this region. You can select multiple instances.

©2013-2026 Tencent Cloud. All rights reserved.

	For all specifications and software subscription fees, see Billing Overview. • When the billing mode is set to Yearly/Monthly Subscription, only CVM instances with this billing mode can be selected. The TI-ONE software subscription fee is prepaid by month. • When the billing mode is set to Pay-as-You-Go, only pay-as-you-go CVM instances can be selected. The TI-ONE software subscription fee is postpaid by hour.
Purchase Duration	• This option becomes available only when the billing mode is set to Yearly/Monthly Subscription. • It indicates the usage duration of CVM instances in TI-ONE.
Auto-Renewal	• This option becomes available only when the billing mode is set to Yearly/Monthly Subscription. • This field is not checked by default.

Note:

1. Precautions before adding CVM instances to TI-ONE:
   - Data
     - Be sure to **back up data** before the CVM instances under your account are selected on TI-ONE.
     - The CVM instance you selected must have a system disk of **at least 100 GB** if no data disk is attached. If a data disk is attached, the data disk size must be **at least 100 GB**. Therefore, we recommend that you mount a data disk before adding the CVM instance to TI-ONE.
   - Network
     - Ensure that the CVM instance you selected is **interconnected** with your storage, such as Cloud File Storage (CFS) and GooseFSx.
     - CVM instances in different VPC networks can be managed and used in TI-ONE. However, you need to ensure that their VPC IP ranges do not **overlap**; otherwise, they cannot be selected for deployment.
2. Impacts after adding CVM instances to TI-ONE:
   - Data
     - TI-ONE will **reinstall a system** for the CVM instance you selected, and the **original data on the CVM instance and the mounted data disk will be cleared**.
     - For the CVM instance you selected, you **no longer have permission to log in to it, nor can you use** Tencent Cloud Automation Tools (TAT). If you log

©2013-2026 Tencent Cloud. All rights reserved.

in to and operate the CVM instance, you shall accept all consequences, such as TI-ONE service exceptions or termination.

- Network
  - TI-ONE will open access to the IP range 9.0.0.0/8 for the security group of the selected CVM instance. Additionally, add outbound/inbound rules to the security group for purchased CVM instances: set the source to the VPC IP range of the CVM instance's network and 9.0.0.0/8, with the protocol port as ALL. **Do not delete** rules for these 2 IP ranges during TI-ONE usage, as this will cause CVM instance exceptions.
  - You can add up to 100 rules to a security group by default. Reserve 2 rules for TI-ONE.

For all notes, see Appendix – Risk Warnings.

4. Verify your purchase information, submit your order, select a payment method, and make the payment.
5. Go to the console to check nodes after the purchase.

   Nodes have the following 9 statuses:

Node Status	Description
Deploying	CVM resources are being initialized on TI-ONE.
Deployment Failed	Failed to deploy CVM resources on TI-ONE.
Running	CVM resources have been successfully deployed on TI-ONE and are in use.
Releasing	CVM resources are being automatically released due to expired usage time on TI-ONE or are being manually released by users.
Released	CVM resources have been automatically released due to expired usage time on TI-ONE or have been manually released by users.
Abnormal	CVM resources are experiencing abnormal operation on TI-ONE.
No Longer Used	CVM resources are no longer used on TI-ONE.
Mounting	The data disk mounted to the CVM resource is being managed for use within TI-ONE.

©2013-2026 Tencent Cloud. All rights reserved.

Under Maintenance	CVM resources have encountered a fault and are under automatic maintenance.

Note

- Some CVM instance types support the No Charge when Shut Down feature. When a CVM instance is shut down, the corresponding TI-ONE software subscription fee stops billing, and the node status changes to Abnormal. When the CVM instance is restarted, the node will automatically change to Running within a few minutes, and billing starts.
- Be aware that after the CVM instance is shut down, it may fail to restart due to insufficient resources.

Operations on Resource Groups

Batch Renewal

After entering a resource group whose CVM instance source is Select from CVM instances, you can click **Batch Renewal** to renew the software subscription fees of multiple selected nodes for the same subscription duration.

Batch Release

After entering a resource group whose CVM instance source is Select from CVM instances, you can click **Batch Release** to release multiple selected nodes. After the release, the software subscription fees will be automatically refunded, and the CVM instances will no longer be supported on TI-ONE.

Deletion

After entering a resource group whose CVM instance source is Select from CVM instances, you can click **Delete** to delete the entire resource group. The deleted resource group cannot be restored.

Note:
A resource group can be deleted only when no nodes are in use within the group. Node statuses include Running, Deploying, Abnormal, Releasing, Under Maintenance, and Mounting.

©2013-2026 Tencent Cloud. All rights reserved.

Operations on Nodes

Mounting a Data Disk

1. In the TI-ONE console, for a node in the Running status without a data disk, when you mount a new data disk for it on the CVM side for the first time, **Mount Data Disk** appears in the operations.
2. When you click **Mount Data Disk**, a double-check pop-up window appears, indicating that mounting a data disk will terminate any running tasks/services on this node.
3. After you click **Confirm**, the node status changes to Mounting. The mounting process is expected to take 1 or 2 minutes, after which the **Mount Data Disk** button will disappear.

If your CVM instance already has a data disk and encounters insufficient hard disk space, you can scale out the CBS. After you scale it out according to the documentation, TI-ONE can automatically use this hard disk.

Renewal

After entering a resource group whose CVM instance source is Select from CVM instances, for nodes in the billing mode of yearly/monthly subscription, you can click **Renew**, select Duration for Renewal, and click **Confirm Renewal** to deduct fees from your account balance.

> ⚠ **Note:**
> 1. This renewal is for the software subscription fee, meaning it renews the usage time of the CVM instance within TI-ONE, but not the resource fees of the CVM instance.
> 2. Nodes in pay-as-you-go mode cannot be renewed.

Enabling/Disabling Auto-Renewal

1. After entering a resource group whose CVM instance source is Select from CVM instances, you can click **Automatic Renewal** for nodes in the billing mode of yearly/monthly subscription. The node will then be automatically renewed for one month's software subscription fee upon expiration.

> ⚠ **Note:**
> 1. This auto-renewal is for the software subscription fee, meaning it renews the usage time of the CVM instance within TI-ONE, but not the resource fees of the CVM instance.
> 2. Nodes in pay-as-you-go mode cannot be renewed.

©2013-2026 Tencent Cloud. All rights reserved.

2. Click **Cancel Auto-Renewal** in the node list. Automatic fee deduction will stop, and an expiration reminder will be sent when nodes expire.

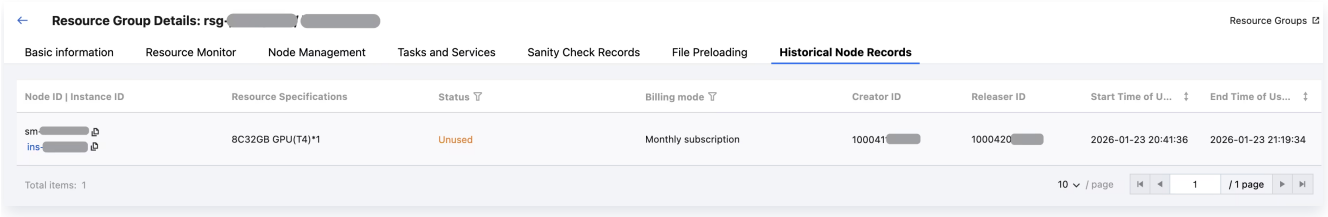
Release

After entering a resource group whose CVM instance source is Select from CVM instances, you can click **Release** to remove nodes in the Running or Abnormal status from TI-ONE usage. The remaining software subscription fee will be automatically refunded after release.

> **Note:**
> This release does not refer to the termination of CVM instance resources but indicates that the CVM instances will no longer be managed on TI-ONE.

Historical Node Records

After entering a resource group, you can click **Historical Node Records** to view the list of nodes in Deployment Failed and No Longer Used statuses.



Appendix

Risk Warnings

1. Precautions before adding CVM instances to TI-ONE:
   - Data
     - Be sure to **back up data** before the CVM instances under your account are selected on TI-ONE.
     - Ensure that the CVM instance selected has a system disk of **at least 100 GB** if no data disk is attached. If a data disk is attached, the data disk size must be **at least 100 GB**. Otherwise, pulling TI-ACC or LLM images will fail, making the CVM instance unavailable for deployment. We recommend that you mount a data disk before adding the CVM instance to TI-ONE.
     - Ensure that for the CVM instance you selected, the highest-priority security group rules (up to 100 rules for inbound direction, 100 rules for outbound direction) have **5**

©2013-2026 Tencent Cloud. All rights reserved.

free rules reserved for TI-ONE to add rules; otherwise, the CVM instance cannot be selected for deployment.

- Network
  - Ensure that the CVM instance you selected is interconnected with your storage, such as CFS and GooseFSx.
  - CVM instances in different VPC networks can be managed and used in TI-ONE. However, you need to ensure that their VPC IP ranges do not overlap; otherwise, they cannot be selected for deployment.
  - If your image has components that need to access the public network, enable **public network access** on the CVM instance (configure the CVM instance with an Elastic IP or ensure the corresponding subnet has a NAT public gateway).
- Supported features
  - The CVM instances selected on TI-ONE support the following capabilities: Task-based Modeling, Notebook, Online Services, and Model Evaluation. **Auto-learning, Visual Modeling, and Batch Predictions are not supported**.

2. Impacts after adding CVM instances to TI-ONE:
   - Data
     - TI-ONE will **reinstall a system** for the CVM instance you selected, and the **original data on the CVM instance and the mounted data disk will be cleared**.
     - For the CVM instance you selected, you **no longer have permission to log in to it, nor can you use** Tencent Cloud Automation Tools (TAT). If you log in to and operate the CVM instance, you shall accept all consequences, such as TI-ONE service exceptions or termination.
   - Network
     - TI-ONE will open access to the IP range 9.0.0.0/8 for the security group of the selected CVM instance. Additionally, add outbound/inbound rules to the security group for purchased CVM instances: set the source to the VPC IP range of the CVM instance's network and 9.0.0.0/8, with the protocol port as ALL. Do not delete rules for these 2 IP ranges during TI-ONE usage, as this will cause CVM instance exceptions.
     - You can add up to 100 rules to a security group by default. Reserve 2 rules for TI-ONE.
   - Finance
     - For CVM instances from your account selected for use in TI-ONE, operations such as renewal, payment, contracts, invoices, refunds, and configuration downgrades are still performed within your account.

©2013-2026 Tencent Cloud. All rights reserved.

Configuration of the Node Network

Last updated: 2026-02-26 16:02:48

After adding purchased CVM instances to the resource group of Tencent Cloud TI-ONE Platform (TI-ONE), you need to configure node networks to ensure proper resource scheduling. This document provides a detailed introduction to how to configure and check network-related settings, including security groups and subnets.

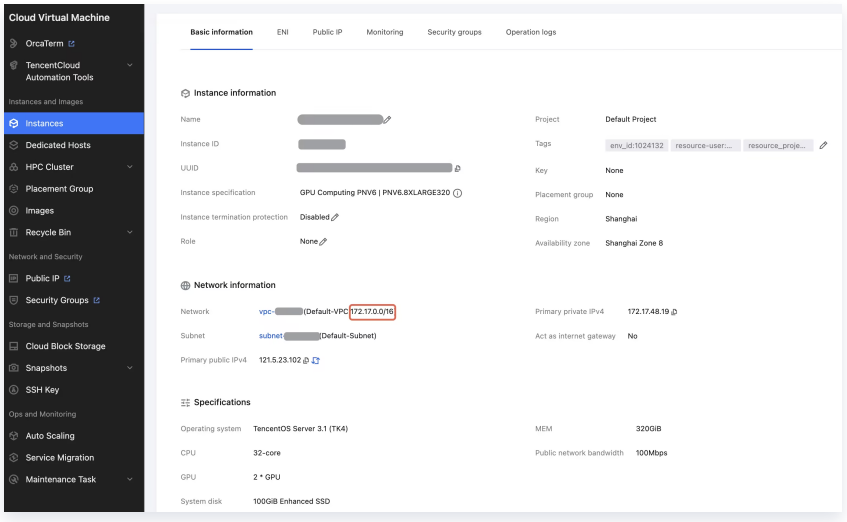
Security Group

A security group is a virtual firewall with stateful data packet filtering features. It is used to set network access controls for instances such as Cloud Virtual Machine (CVM), Cloud Load Balancer (CLB), and cloud database. It controls inbound and outbound traffic at the instance level and is an important means of network security isolation.
When using TI-ONE, you need to manually open 2 security group IP ranges for CVM instances, including the Virtual Private Cloud (VPC) IP range to which the CVM instance belongs and 9.0.0.0/8. You can add up to 100 rules to a security group by default. Please reserve 2 rules for TI-ONE.

Must-Knows

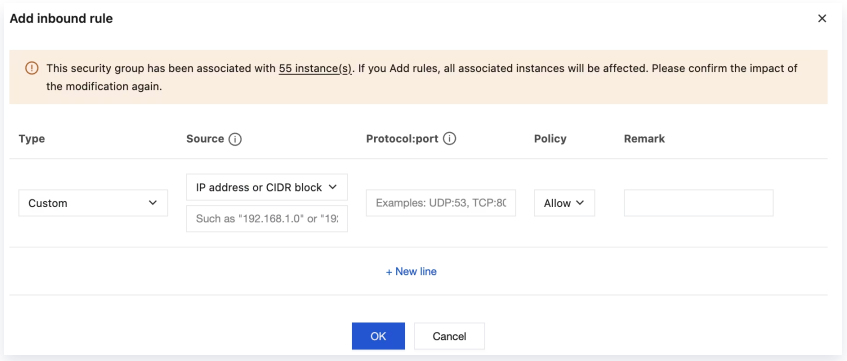
After purchasing CVM instances, add the corresponding VPC IP range to which the CVM instance belongs and 9.0.0.0/8 to the security group, and configure inbound and outbound rules to ensure normal communication during training/inference. Do not delete rules for these 2 IP ranges during TI-ONE usage, as this will cause CVM instance exceptions.
The specific operation guide is as follows:

1. Purchase a CVM instance and obtain the **Network** information on the instance details page. (Remarks: this corresponds to the VPC IP range to which the CVM instance belongs.)



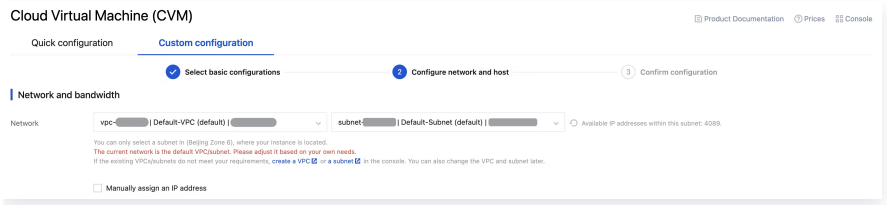
©2013-2026 Tencent Cloud. All rights reserved.

2. On the **Security Group Details>Security Group Rules**tab page, add two inbound and two outbound rules.
   - **Source**: Set to the VPC IP range in the Network field of the previous step and 9.0.0.0/8.
   - **Protocol Port**: Set to ALL.



Subnet

Subnet is an IP address block within each VPC network, and all cloud resources in the VPC network must be deployed within subnets.
When TI-ONE launches a task/service, each Pod occupies one subnet IP address of the CVM instance. If there are insufficient IP addresses, node addition may fail, or tasks/services may not start normally after the node is added. Therefore, you need to select a VPC subnet with sufficient IP addresses when purchasing CVM instances.



Must-Knows

When you add CVM instances to a resource group, TI-ONE verifies whether the VPC subnet to which the selected nodes belong has sufficient IP addresses:

1. **Adding nodes prohibited when available IP addresses are insufficient**
   - Verification condition: If the number of remaining available IP addresses in the VPC subnet to which the selected nodes belong is less than the number of selected nodes.
   - Effect: You cannot click **Buy Now**, and adding nodes is not allowed.
   - Example: Suppose 5 nodes are selected, with Node_1, Node_2, and Node_3 using Subnet_A, and Node_4 and Node_5 using Subnet_B. The platform then verifies whether the number of remaining available IP addresses (n) in Subnet_A is less than 3 and whether the number in Subnet_B is greater than or equal to 2.
2. **Alarm prompted when there are few available IP addresses**

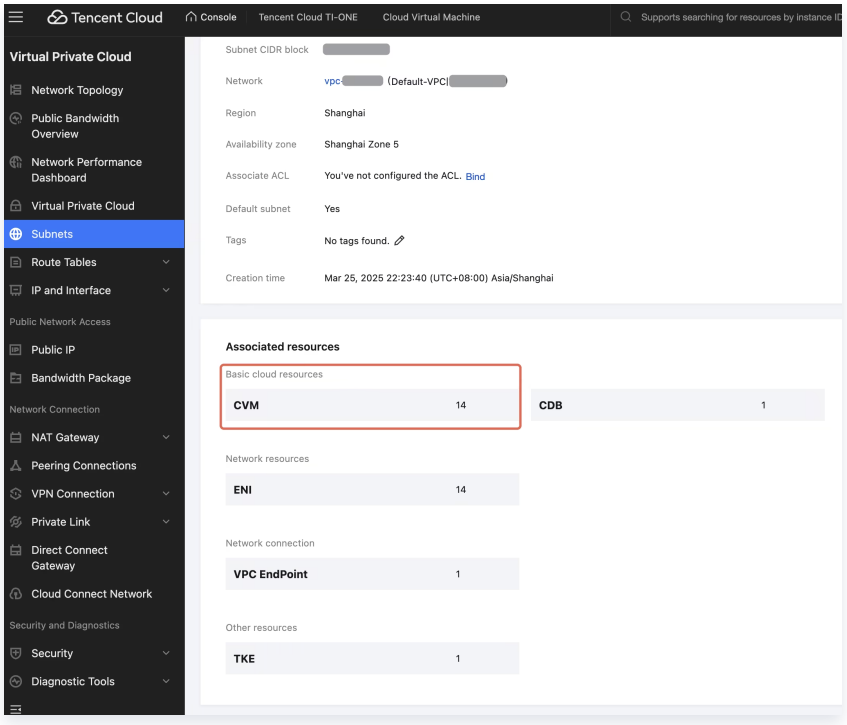
©2013-2026 Tencent Cloud. All rights reserved.

- Verification condition: If the number of remaining available IP addresses in the VPC subnet to which the selected nodes belong is greater than or equal to the number of selected nodes, but less than the number of selected nodes x 10.
- Effect: Adding nodes is allowed, but the platform will prompt that there are few available IP addresses in the corresponding subnet.
- Example: Suppose 5 nodes are selected, with Node_1, Node_2, and Node_3 using Subnet_A, and Node_4 and Node_5 using Subnet_B. The platform then verifies whether the number of remaining available IP addresses (n) in Subnet_A falls within the interval 3 ⩽ n< 30, and whether the number in Subnet_B falls within the interval 2 ⩽ n < 20.

Handling Measures

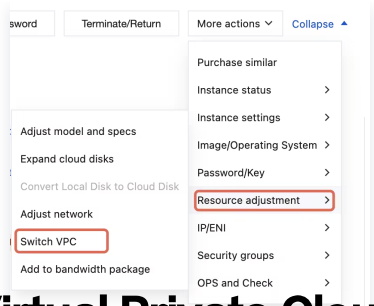
When the available IP addresses in a subnet are insufficient, you can go to the CVM console to switch VPC networks. The procedure is as follows:

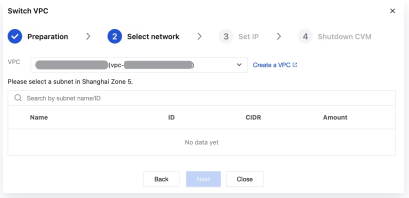
1. If you encounter the above situation when adding nodes, you can click the name of the VPC subnet in the prompt to redirect to the VPC console and go to the details page of the corresponding subnet.
2. On the subnet details page, click **CVM** to redirect to the instance list and view the CVM instances using this subnet.



3. Locate the CVM instance to be added to TI-ONE, go to the instance details page, choose **More Actions** > **Resource Adjustment** > **Switch VPC** in the upper-right corner, and modify the VPC network and subnet in the pop-up window.

©2013-2026 Tencent Cloud. All rights reserved.





Virtual Private Cloud

VPC is a dedicated network space built on Tencent Cloud, providing logically isolated network services for your resources on Tencent Cloud (such as CVM).
After adding and managing CVM instances on TI-ONE, if these CVM instances reside in different VPC networks, you need to use Cloud Connect Network (CCN) to establish connections, ensuring that CVM instances across multiple VPC networks can communicate properly.

Must-Knows

1. **VPC planning and connectivity**
   - If you need to manage CVM instances in multiple VPC networks within the same region, plan the number and IP ranges of VPC networks based on business requirements, ensuring that the IP ranges of each VPC network do not overlap to avoid network conflicts. For specific planning methods, see Network Planning.
   - Meanwhile, if there is a need for cross-VPC communication, you can establish connections between VPC networks in advance via the CCN console.
   - Example: Suppose you add CVM instances N1 and N2 to the resource group in TI-ONE, where N1 belongs to the VPC IP range 10.0.0.0/12 and N2 to 10.16.0.0/12. You can establish the connection between these two IP ranges in CCN. For specific operations, see Handling Measures below.
2. **Network connectivity probe**
   - After a CVM instance is added to the platform, the platform sets the VPC network to which the first node joining the resource group in the region belongs as the default VPC network. In addition, the platform periodically checks the network connectivity status between the VPC networks of other CVM nodes and the default VPC network to avoid node unavailability caused by network inaccessibility.
   - Display effect: If network inaccessibility is detected, a highlighted tag Abnormal Network Connection will appear below the IP address field in the corresponding resource group node list. Hover the mouse pointer to view the operation guide and specific VPC information, helping you quickly locate and resolve network issues.
3. **Endpoint description**

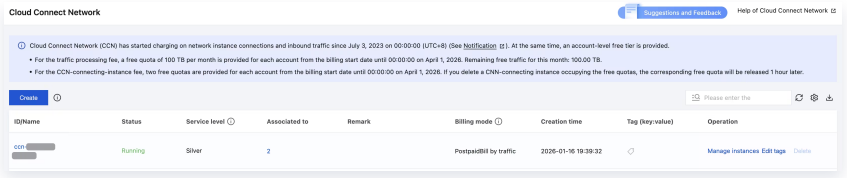
©2013-2026 Tencent Cloud. All rights reserved.

- To ensure stable communication between CVM nodes and system components of the platform, the platform will automatically create an endpoint named "tione" on the Endpoints page in the VPC console after you add a CVM instance for the first time. For details, see Cross-Account VPC Service Sharing.
- Note: Do not manually delete this endpoint during platform usage, as it may cause abnormal CVM node features or prevent normal use of platform services.
- Example: Suppose you add CVM instances N1 and N2 to the resource group in TI-ONE, where N1 belongs to the VPC IP range 10.0.0.0/12 and N2 to 10.16.0.0/12.

Handling Measures

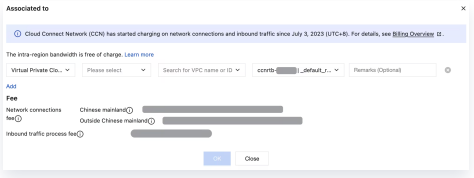
When VPC networks between different nodes are not connected, or when a node's VPC network cannot communicate with the platform's default VPC network, you can go to the CCN console to establish the connection. For details, see Connecting Network Instances Under the Same Account. A process example is as follows:

1. Create an instance: In the CCN console, click **Create** for CCN instance creation.

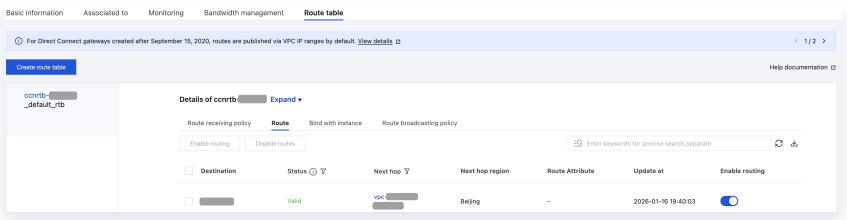


2. Associate VPC: Go to the instance details page, and click **Add Instance** in the Associate Instance tab. Add the VPC networks that need to be connected, for example, the VPC networks of nodes N1 and N2, or the VPC network of node N3 and the platform's default VPC network.





3. Check route table: In the Route Table tab, check whether the routing policies of each subnet under VPC associated with CCN have taken effect. If the IP ranges of the associated network instances conflict, invalid routes will be generated.

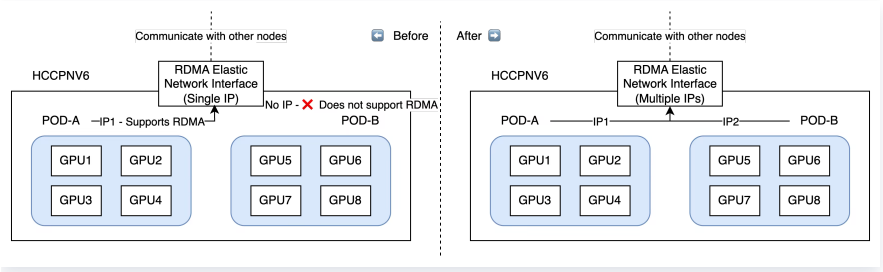


Multi-IP Addresses RDMA ENI

©2013-2026 Tencent Cloud. All rights reserved.

> **Note:**
> This feature is in open beta. If you have testing needs, contact the product manager to enable the allowlist.

By default, each high-performance CVM instance type supporting remote direct memory access (RDMA) has only one IP address on its Elastic Network Interface (ENI). If a 4-GPU card Pod runs on a node with a total of 8 GPU cards, this Pod will occupy the only IP address of the RDMA ENI on the node. Consequently, other Pods running on the node cannot use the RDMA ENI for communication, as shown in the following figure:



If you need to support RDMA communication between multiple Pods on a single node, you need to enable the multi-IP addresses RDMA ENI feature.

- The currently supported instance types include HCCPNV6, HCCPNV6c, HCCPNV6d, and HCCPNV6s. (All of these are HCCPNV6 series instance types supporting RDMA.)

Handling Measures

1. Contact the CVM team to apply for activating the "28-bit mask for RDMA IP addresses" allowlist and provide the region of the node and the ID of the high-performance cluster (which can be viewed in the CVM console).
2. After the allowlist is activated, TI-ONE will detect whether the nodes support the multi-IP addresses RDMA ENI capability and display it via highlighted tags in the node list on the resource group details page.
3. When creating a training task or inference service, select the resource group where the above nodes are located, and enable the Use RDMA configuration. For details, see Creating a Task and Online Services Deployment.
   3.1 For training tasks: When configuring full GPU card and selecting more than 1 node, you can configure whether to enable RDMA. RDMA is enabled by default for multi-node tasks configured with full 8 GPU cards per node. When RDMA is enabled, tasks are preferentially scheduled to nodes that support RDMA. When there are insufficient IP addresses for RDMA ENIs on a single node, tasks may be scheduled to multiple nodes.
   3.2 For inference services: When the number of replicas or the number of instances per replica exceeds 1, and the selected GPU model supports RDMA, you can enable RDMA.

©2013-2026 Tencent Cloud. All rights reserved.

After RDMA is enabled, services are preferentially scheduled to nodes supporting RDMA. If a single node lacks sufficient IP addresses for RDMA ENIs, services may be scheduled to multiple nodes.

©2013-2026 Tencent Cloud. All rights reserved.

Scheduling and Quota Management

Last updated: 2026-02-26 16:03:05

1. Scheduling Policy

1.1. Overall Description

Tencent Cloud TI-ONE Platform (TI-ONE) resource groups support the queuing policy and priority scheduling policy when scheduling training tasks and online services.

- Queuing policy: When resources are insufficient, the default queuing policy follows the First-In-First-Out (FIFO) order based on the submission time of tasks/services. You can also configure a traversal policy in the resource group to prioritize scheduling tasks/services in the queue that meet resource requirements. For details, see Queuing and Preemption Policy Configuration.
- Priority scheduling policy: You can set the priority of tasks/services based on tags (with P0 as the highest, and P9 as the lowest). The default priority preemption policy is set to allow high-priority tasks/services to preempt low-priority tasks. You can also configure the priority preemption policy by functional module based on actual conditions. For detailed configuration methods, see Queuing and Preemption Policy Configuration.
  - Remarks: The feature of setting task priority through tags is only supported in the Task-based Modeling and Online Services modules.

Note:

- CPU tasks are evenly distributed to different nodes of the resource group by default according to the Cloud Load Balancer (CLB) policy.
- GPU tasks are preferentially scheduled to the same node in a way that minimizes fragments by default.
- Tasks of different GPU card types do not block each other. For example, tasks on T4 cards do not affect tasks on A100 cards.

1.2. Queuing and Preemption Policy Configuration

The initial policy statement for the resource group is as follows:

```
{
  "Version": "1.0",
  "ResourceRule": {
    "DefaultPriority": 0,
    "DefaultQueue": 0,
    "Preempted": 1
```

©2013-2026 Tencent Cloud. All rights reserved.

```
  },
  "TaskRules": []
}
```

1.2.1. Parameter Description

- **DefaultPriority:** Default priority. The priority range is 0-9, where P0 is the highest. The default value is 0.
- **DefaultQueue:** Default queuing policy. Available values: 0, 1, and 2. Default value: 0. This is only supported in the Task-based Modeling and Online Services modules.
  - 0 represents strict FIFO. Regardless of how many resources are currently available, the earliest submitted task in the waiting queue is selected. If resources are sufficient, it is scheduled for execution; otherwise, it waits.
  - 1 indicates a traversal policy within the priority level, which means that tasks of the same priority may not strictly dequeue in the order of task submission time and can jump the queue based on resource availability.
  - 2 indicates a traversal policy for all queues, which means that the entire waiting queue follows the traversal policy, and tasks are dequeued as long as resource requirements are met.
- **Preempted:** Default policy for priority preemption. Available values: 0 and 1. The default value is 1.
  - 0 means high-priority tasks in the queue will not preempt low-priority tasks that are already running.
  - 1 means high-priority tasks in the queue will preempt low-priority tasks that are already running.

1.2.2. Configuring Preemption by Module

If you need to configure preemption policies by module, currently, you can set preemption policies separately for Task-based Modeling and Online Services modules. As shown in the following statement, you need to add the **PreemptedRule** field and assign values to **TrainingTask** (Task-based Modeling) and **ServiceGroup** (Online Services). Setting it to **1** means tasks or services in this module support preemption, while setting it to **0** means they do not. When both **PreemptedRule** and **Preempted** exist, the rules set in **PreemptedRule** take precedence.

```
{
  "Version": "1.0",
  "ResourceRule": {
    "DefaultPriority": 0,
    "DefaultQueue": 0,
    "Preempted": 1,
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    "PreemptedRule": {
      "TrainingTask": 1,
      "ServiceGroup": 0
    }
  },
  "TaskRules": []
}
```

1.3. Guide for Priority Configuration

1.3.1. Example of Policy Syntax

At the resource group level, you can configure a policy description file for the **scheduling policy**. Here, **DefaultPriority indicates the default priority. In TaskRules, priorities can be set via task tags as shown in the following example, where ValueType** defaults to **Tag.** ValueType may be supplemented based on actual requirements in the future.
Below is an example statement that implements this scenario: Tasks/Services under this resource group have a default priority of P0. If the tag of a task/service matches **Task Type – Voice Task**, then the priority for that task/service is set to P1. (Note: The feature of setting task priority via tags is supported only in Task-based Modeling and Online Services modules.)

> **Note:**
> The # comment sections in the following policy statement do not comply with JSON syntax and are for documentation display only. They must be removed before the policy statement is pasted into the platform.

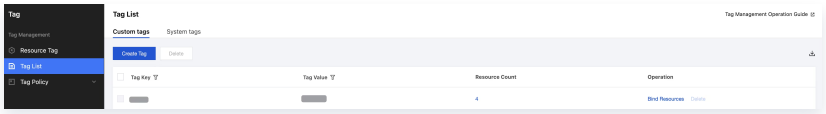
```
{
 "Version":"1.0", // Version in JSON format.
 "ResourceRule":{
   // Policy for the resource group.
   "DefaultPriority": 0, // Default priority. Priority range: 0-9.
   "DefaultQueue": 1,
   "Preempted": 1
 },
 "TaskRules":[
   // Task scheduling policy, a list of matching rules
   {
     // Each rule defines a relationship between properties and
scheduling
     "AttrRules":[{
       "ValueType":"Tag",
```

©2013-2026 Tencent Cloud. All rights reserved.

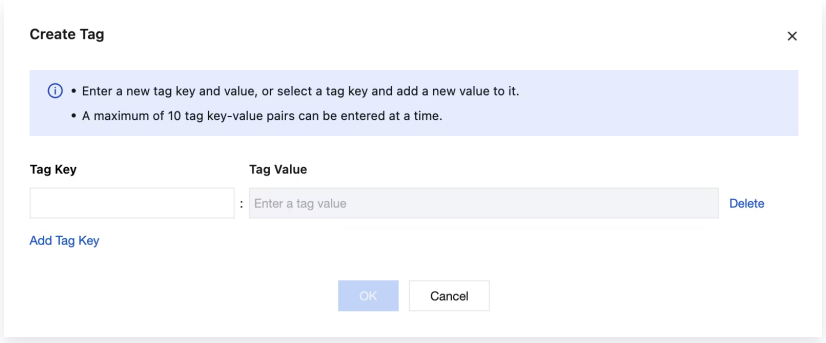
```
          "Key":"Task Type",
          "Value":"Voice Task"
        }],
        "ScheduleRule"
:{          "Priority": 1 // Set priority. Priority range: 0-9
        }
      }
    ]
}
```

1.3.2. Specific Operation Process (Taking Task-based Modeling as an Example)

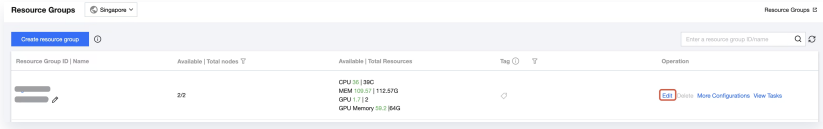
1. Log in to the Tag console, and go to the **Tag List**.



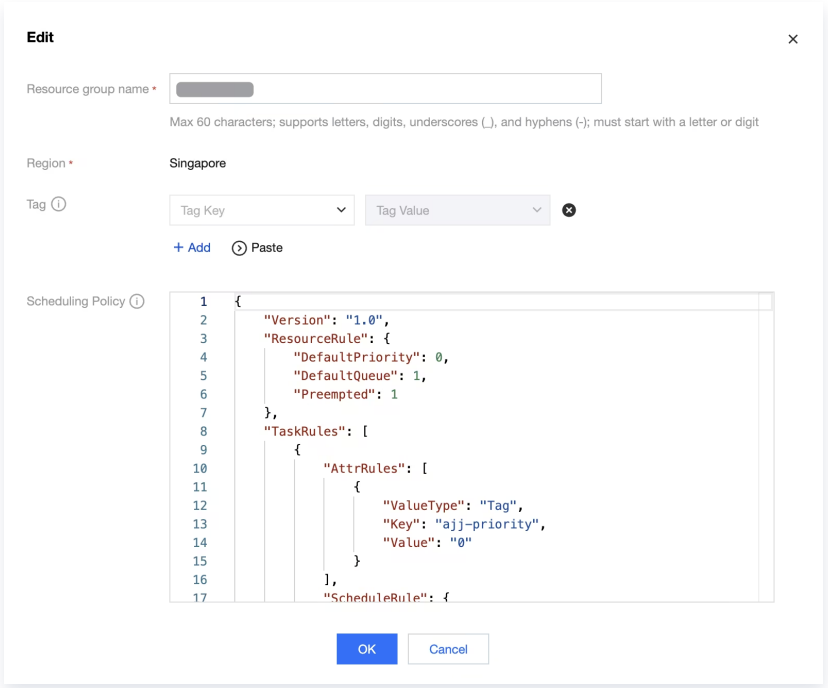
Click **New Tag**, enter the tag key **Task Type** and tag value **Voice Task**.



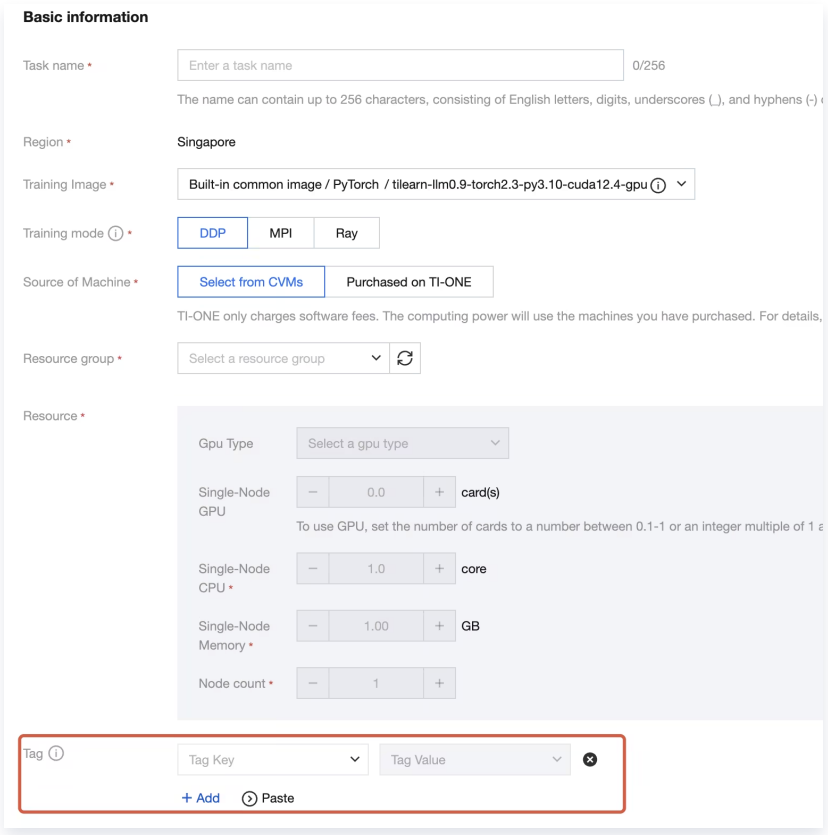
2. Go to the TI-ONE resource group management page, select a resource group. Click **Edit**, and enter the above policy description statement in the **Scheduling Policy** column.



©2013-2026 Tencent Cloud. All rights reserved.



3. When a new training task is created, tag this task with **Task Type** > **Voice Task** during task creation if you need to set this task to low priority. Then, during scheduling, this task will be treated as a low-priority task, can be preempted, and will automatically enter the waiting queue.

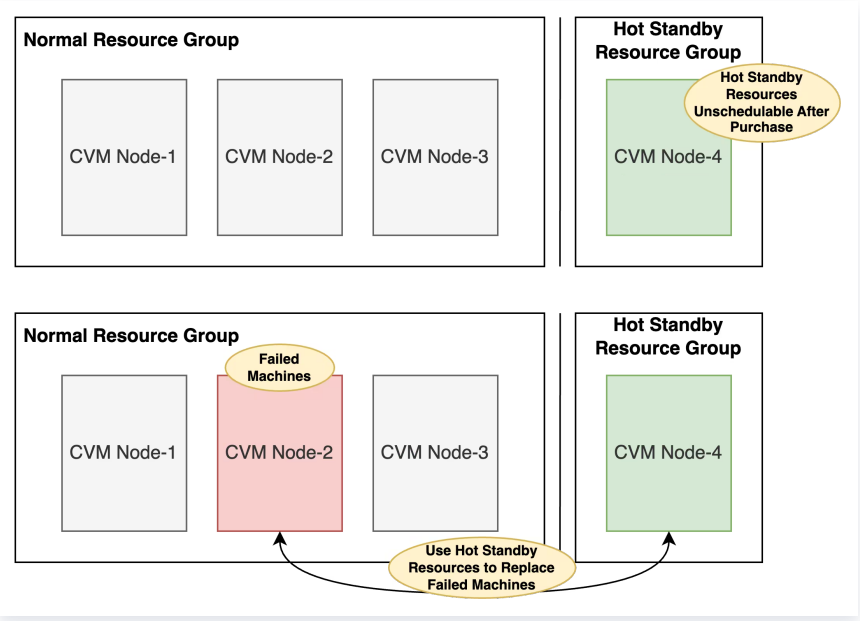


2. Hot Standby Resource Management

©2013-2026 Tencent Cloud. All rights reserved.

> **Note:**
> This feature is in open beta. If you have testing needs, contact the product manager to enable the allowlist.

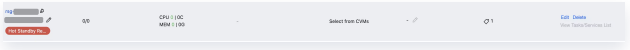
TI-ONE resource groups support enabling the Hot Standby Resource Group feature. When a CVM instance experiences an exception causing tasks/services to fail to run normally, the platform will move available nodes from the hot standby resource group to the resource group where the abnormal node is located, ensuring rapid recovery of tasks/services. The process is shown in the figure below:



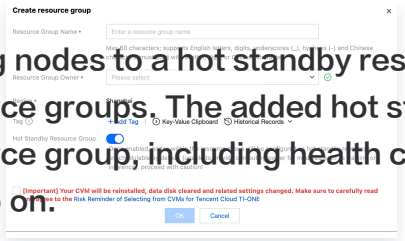
2.1. Configuring a Hot Standby Resource Group

1. In the New Resource Group pop-up window, enable the switch to configure the resource group as a hot standby resource group. After creation, the hot standby resource group will be displayed with a highlighted tag in the resource group list.

> **Note:**
> - Note: Only one hot standby resource group can be created per region.
> - Since hot standby resources are not schedulable by default, the hot standby resource group does not require configuration of scheduling policy-related parameters.



©2013-2026 Tencent Cloud. All rights reserved.



2. Adding nodes to a hot standby resource group follows a process identical to that for regular resource groups. The added hot standby nodes also support all other features within the resource group, including health checks, file preloading, adding/releasing/moving/renewing, and so on.

> **Note:**
> CVM instances in the hot standby resource group need to be purchased separately. Please evaluate carefully.

2.2. Description of Hot Standby Logic

1. **Not schedulable hot standby resources**

   Nodes in the hot standby resource group serve as hot standby resources and are not schedulable by default (unable to provide computing resources for training/inference modules). Therefore, when you create tasks/services in modules such as Task-based Modeling, Dev Machines, and Online Services, Hot Standby Resource Group will not appear in the resource group dropdown list.
2. **Trigger for replacing with hot standby resources**

   When a CVM instance requires maintenance, or the node status changes to Abnormal (such as GPU card failure) and persists for 1 minute, rendering the node completely unavailable, the platform will replace the faulty CVM instance with hot standby resources.
3. **Specific operations for replacing with hot standby resources**

   When a faulty CVM instance is replaced, the platform identifies a CVM instance with the same instance specifications as the abnormal node in the hot standby resource group, and explicitly moves it to the resource group containing the abnormal node. Meanwhile, the repaired faulty CVM instance (successfully maintained and restored to an available state) will be automatically moved back to the hot standby resource group.

3. Managing Computing Power Quotas

> **Note:**
> This feature is in open beta. If you have testing needs, contact the product manager to enable the allowlist.

TI-ONE resource groups support enabling the User Quota Management feature to configure the upper limit (quota) of resources available to specified users within the resource group. The following sections describe how to configure computing power quotas and restrict usage based on these quotas.

3.1. Configuring Quotas

©2013-2026 Tencent Cloud. All rights reserved.

1. On the Resource Group Details page, enable the User Quota Management feature. Once enabled, you can configure both Default Quota and Custom Quota.
2. Default quota: indicates the default upper limit (quota) of GPU/CPU/memory resources available for each user using the current resource group. Users without a custom quota will use the default quota.
3. Custom quota: indicates the upper limit (quota) of GPU/CPU/memory resources available for designated users when tasks/services are created using the current resource group. You can add Tencent Cloud sub-accounts or user groups via dropdown options and allocate the maximum available number of GPU cards/CPU cores/memory to them.
   - Remarks: A maximum of 50 custom quotas are supported.
4. The relationship between default quotas and custom quotas: Suppose users A, B, and C can access the resource group. If custom quotas are configured for users A and C, then the resources available for A and C will not exceed their custom quotas, while the resources available for B will not exceed the default quota.
5. A custom user group quota indicates that the quota applies to each user within the group individually, rather than being shared by all users in the group. For example, if a user group includes users A, B, and C, and a custom quota of 32 cores, 128 GB, and 8 GPU cards is configured for this group, the upper limit of resources available for each of users A, B, and C will be 32 cores, 128 GB, and 8 GPU cards.

3.2. Logic for Quota Verification

1. **Verification during creation**

When a task or service is created in modules like Task-based Modeling, Dev Machines, Online Services, and Model Evaluation, the system verifies whether the amount of resources currently requested by the user exceeds the total quota. If the amount does not exceed the quota, the task or service can be created normally; if the amount exceeds the quota, an error will be reported, and the operation will be blocked. For example:

- Suppose that in a certain resource group, the quota for User A has been configured as 32 CPU cores, 128 GB of memory, and 8 GPU cards.
- When User A creates a Task-based Modeling task requesting 16 cores, 64 GB, and 2 T4 GPU cards, the task can be submitted normally. However, a task requesting 64 cores, 256 GB, and 16 T4 GPU cards will be blocked. During creation, the system will report an error: "Requested resources exceed the user's maximum quota in the current resource group: 32 cores, 128 GB, and 8 GPU cards. Please adjust resources and retry, or contact the resource group owner."

2. **Verification during scheduling**

After a task or service is submitted, if the Currently Requested Resources exceed the User Quota - Used Quota (Running Resources), the task will enter the waiting queue. This aims to maximize resource utilization and avoid idling. For example:

©2013-2026 Tencent Cloud. All rights reserved.

- Suppose that in a certain resource group, the quota for User A has been configured as 32 CPU cores, 128 GB of memory, and 8 GPU cards.
- First, User A creates a training task requesting 16 cores, 64 GB, and 2 T4 GPU cards. After submission, the task runs normally (remaining available resources at this point: 16 cores, 64 GB, 6 GPU cards).
- Next, User A creates an online service requesting 32 cores, 128 GB, and 8 T4 GPU cards (which does not exceed the quota). This can be submitted normally, but after submission, it enters the waiting queue.
  - The service status shows as Queued with a prompt: Requested resources exceed the remaining available quota for the current user in the resource group. Please stop other tasks/services or contact the resource group owner.</n>Maximum available: 32 cores, 128 GB, and 8 GPU cards.
- After the preceding task completes and releases resources, they can be used to schedule the online service.

3. **Effective logic when a user has multiple quotas**

Due to the existence of user group quotas, a user may have multiple quotas simultaneously. In this case, the effective quota is determined by taking the maximum value for each parameter among all applicable quotas. For example:

- Suppose a custom quota of 16 cores, 32 GB, and 2 GPU cards has been added for User A; in addition, User A belongs to a user group with a quota of 32 cores, 64 GB, and 1 GPU card.
- When end user A creates a task or service, the upper limit of requestable resources will be 32 cores, 64 GB, and 2 GPU cards.

©2013-2026 Tencent Cloud. All rights reserved.

File Preloading

Last updated: 2026-02-26 16:03:18

Overview

File preloading involves caching selected files onto the nodes of a specified resource group in advance to enhance the startup speed and stability of training tasks/online services (in LLM training scenarios, the time required to load files into Pod memory can be reduced from 2 hours to 5 minutes). This document primarily introduces how to add files and preload them into resource groups, as well as how to mount cached files in modules such as training or inference.

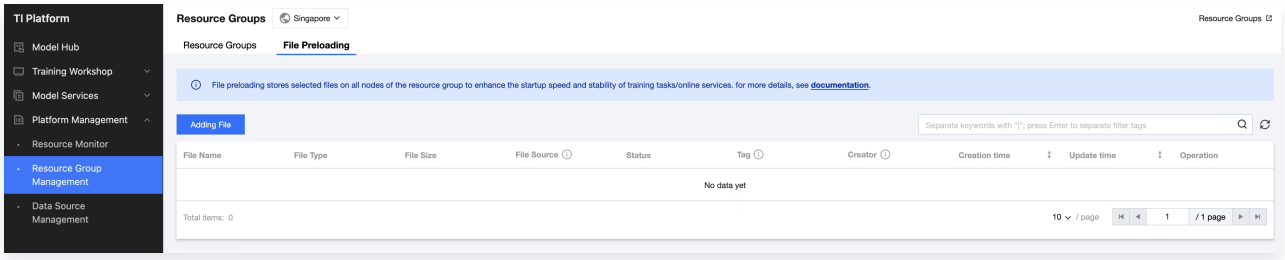
Prerequisites

1. **Resource group created.**

   Choose Console > Resource Group Management to create a resource group in advance and add nodes. For the operation process, see Creating a Resource Group.
2. **Cloud Object Storage (COS) activated.**
   - The file preloading feature only supports adding files from built-in models or storage instances of COS type. Go to Console > Cloud Object Storage to activate the storage service according to interface prompts. For the operation process, see Getting Started.
   - COS: COS is a distributed storage service provided by Tencent Cloud for storing massive files, with advantages such as high scalability, low cost, reliability, and security. Through diverse methods such as console, API, SDK, and tools, you can easily and quickly access COS to upload, download, and manage files in multiple formats, achieving massive data storage and management.

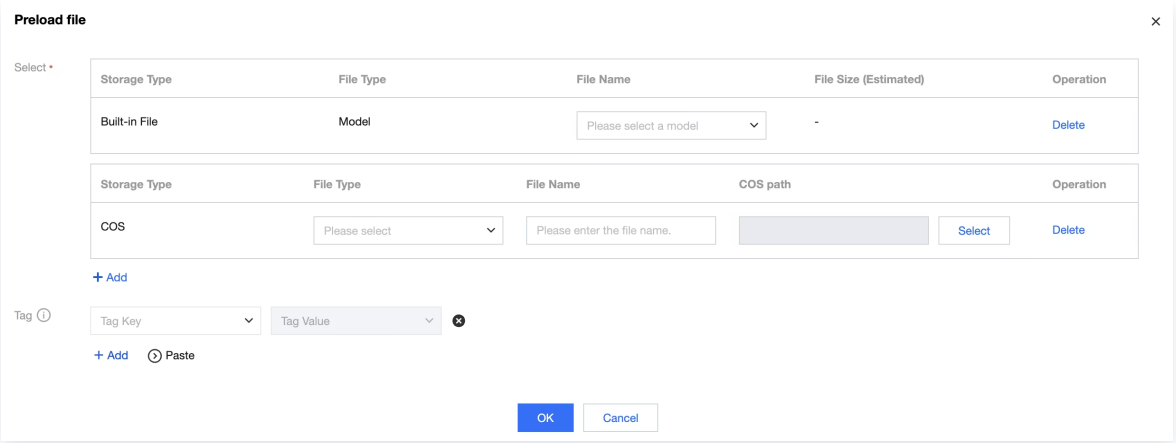
Adding a File

1. In the left sidebar, choose **Platform Management > Resource Group Management**, and go to the File Preloading page. You can add cached files to all resource groups and load files to resource groups at a global level.



2. Click **Add File**, and a pop-up window as follows will appear. You can select files from **Built-in Model** and **COS** storage types.

©2013-2026 Tencent Cloud. All rights reserved.



Configuration parameters and their descriptions are listed in the table below.

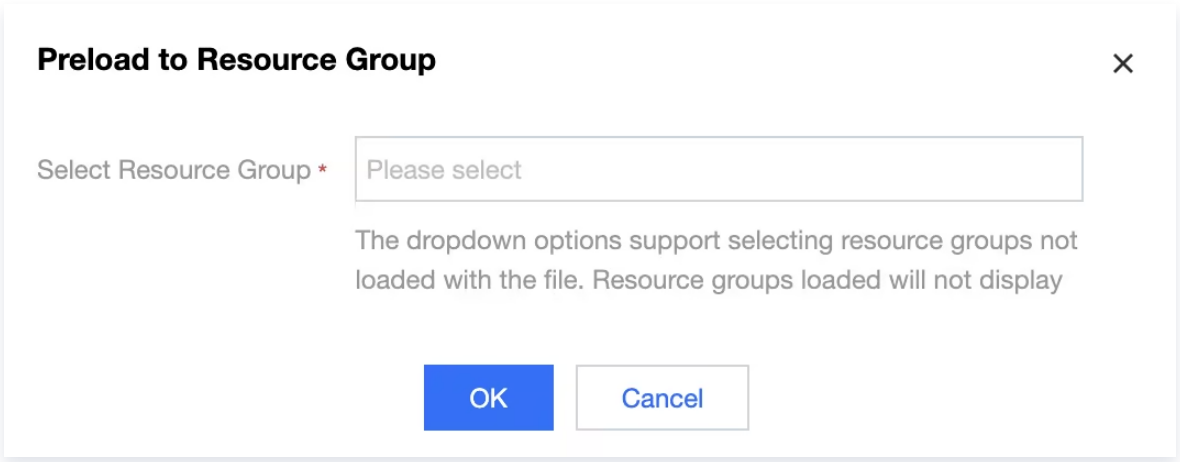
Storage Type	Field Name	Description
Built-in File	File Type	Indicates the type tag of the preloaded file. It is used to identify the file's type and purpose when mounts are added to Task-based Modeling/Online Services. (Remarks: Currently, only built-in files of the Model type are supported. Future versions will gradually support file objects such as images and data sets.)
	File Name	Indicates the built-in file objects supported by the platform for preloading. Required. The options are sourced from models listed in the Model Hub.
	File Size	Indicates the estimated size of disk space occupied by built-in files.
COS	File Type	Used only to mark the type of preloaded files. The options include models, images, and data sets.
	File Name	• Indicates the definition and description of preloaded files. • Required. Only Chinese characters, English letters, digits, underscores (_), and hyphens (-) are allowed. It must start with a Chinese character, English letter, or digit, with a maximum length of 256 characters.
	COS Path	Indicates the specific path of the file source. Only supports selecting folders; does not support selecting specific files.

3. After selecting the file, click **Confirm**. The added file will be displayed in the list.

Preloaded Files

©2013-2026 Tencent Cloud. All rights reserved.

1. Click **Preload** in the Operation column of the list. You can load the selected file to multiple resource groups.



2. After selecting a resource group, click **Confirm**. The platform will execute the loading task on each node within the selected resource group.

Managing Files

In the list on the File Preloading page, you can view the file loading status at a global level, along with loading details for each resource group. Additionally, operations such as updating or deleting files are supported.

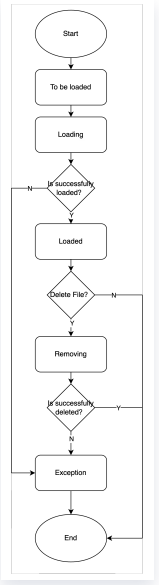
Node Status

After a file is added, the initial status of new data in the list is **Pending**. The status will only change after the file is added to any resource group, triggering the subsequent status transition: **Loading** > **Loaded/Partially Loaded/Abnormal** > **Removing**. The status descriptions are as follows:

Status	Description	Status Transition

©2013-2026 Tencent Cloud. All rights reserved.

Pending	Indicates that the file has not been loaded on any resource group yet. **Note:** Currently, since the file has not been loaded, you cannot click **Update**.
Loading	Indicates that the file is being loaded on all nodes in the specified resource group. This status is displayed only when all resource groups are in the Loading status.
Abnormal	• Indicates that an exception occurred during the file loading or removal process. This status is displayed only when all resource groups are in the Abnormal status. • Hover over the icon on the right to display detailed error messages. When multiple resource groups encounter loading exceptions, the exception information of the first resource group is displayed.
Loaded	• Indicates that the file has been successfully loaded on all nodes across all resource groups. This status is displayed only when all resource groups are in the Loaded status. • Hover over the icon on the right to display the names of all loaded resource groups. Click **View Details** to view the complete list.
Partially Loaded	Indicates that the file has been successfully loaded on some resource groups, but some resource groups are abnormal or still loading. This status is displayed when any resource group is in the Loaded status.
Removing	Indicates that the file is being removed from all nodes within the resource group. This status is displayed when any node is in the Removing status. After the file is removed, the data is synchronously deleted from the list. **Note:** Since the file is being removed, you cannot click **Delete** again.

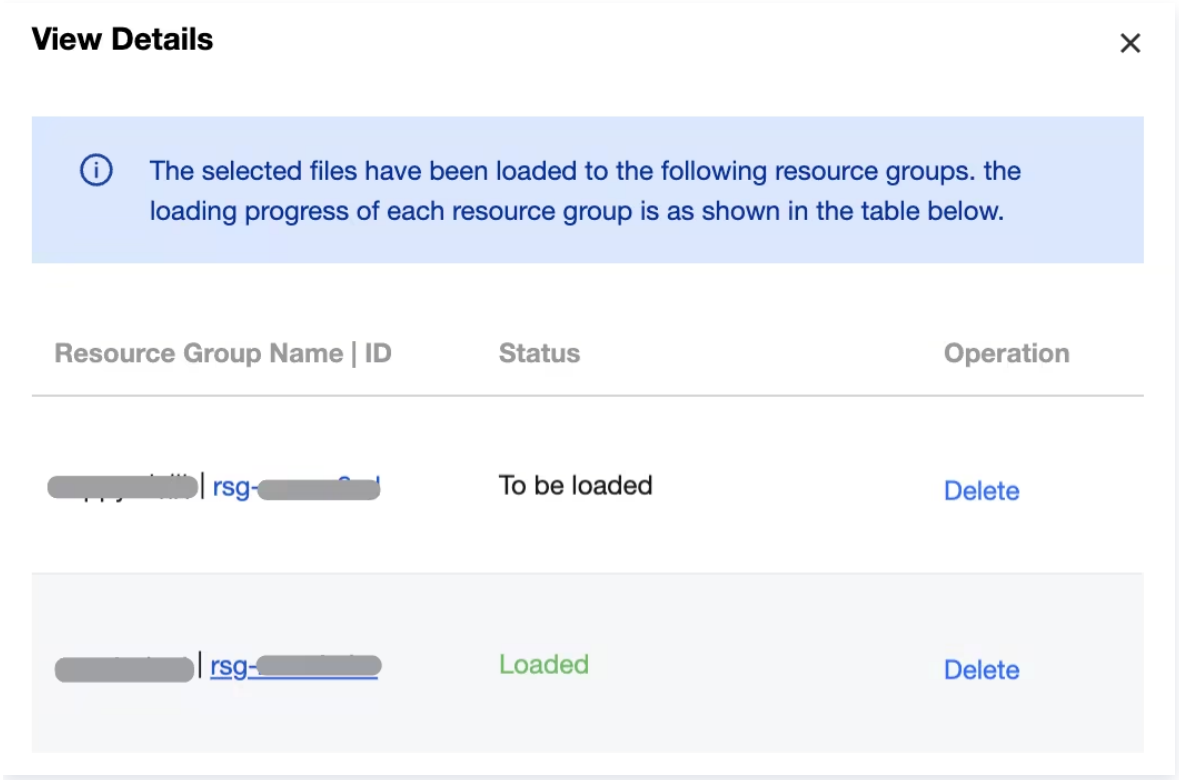


Viewing Details

©2013-2026 Tencent Cloud. All rights reserved.

Click a **file name** in the list to open a pop-up window showing the details of all resource groups loading this file, including the resource group name/ID, loading status, and progress.

- Click a **resource group ID** to redirect to the File Preloading tab on the details page of the specified resource group.
- Click **Delete** to remove the selected file from a single resource group.

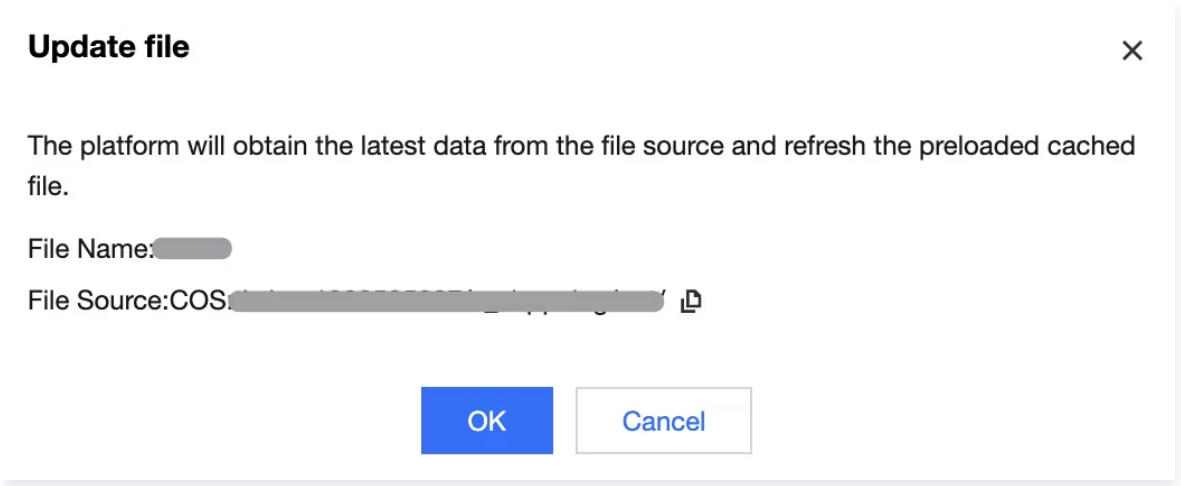


Updating

Click **Update** in the Operations column of the list to obtain the latest data from the source path and update the preloaded cached files.

- Note: The update operation follows an incremental update logic. After the update is triggered, the file will revert to the Loading status (same as the process during initial creation).
- Example of incremental update: Suppose 8 files are loaded in a directory. Then, 4 files are deleted, and 3 files are added. After the incremental update, there should be 7 files (not 11).

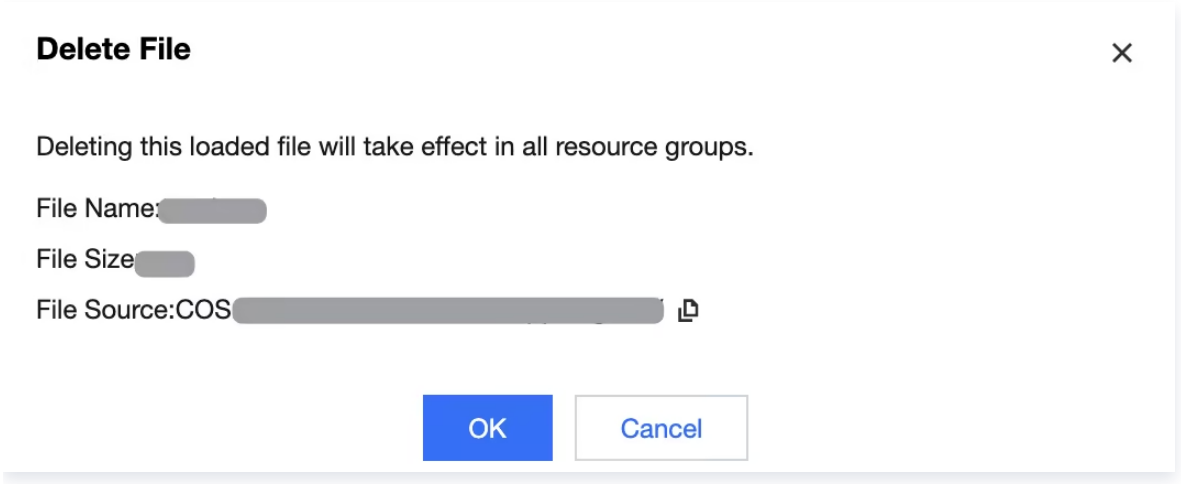
©2013-2026 Tencent Cloud. All rights reserved.



Deleting

Click **Delete** in the Operations column of the list to delete the file from all resource groups where it has been loaded.

> **Note:**
> After you click **Confirm**, the node will enter the **Removing** status, and the file information will be deleted from the list.



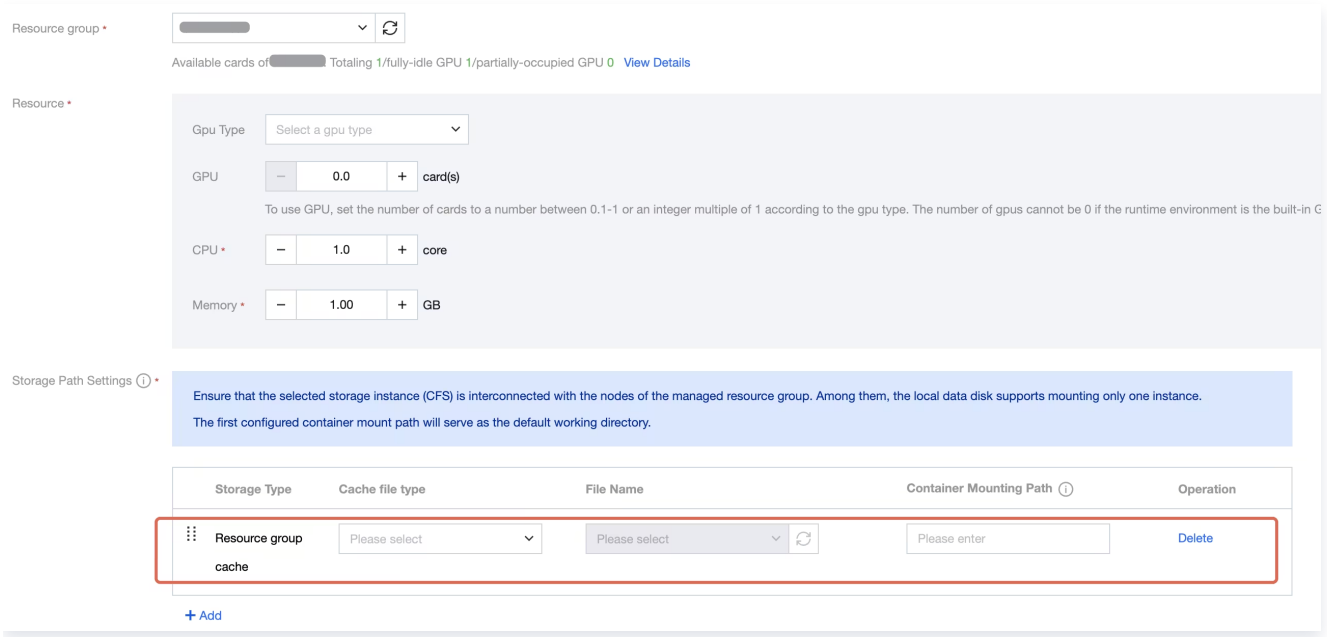
Using Preloaded Files

When you create tasks or services in modules like Task-based Modeling, Dev Machines, Online Services, or Model Evaluation, you can select the mount type **Resource Group Cache** to mount preloaded files from the resource group to the container of the task/service. The following uses Dev Machines as an example: Assume that the model file "**model_file**" is preloaded in resource group A.

1. Go to the Dev Machines module, click **New**, select resource group A on the creation page, and configure the Storage Path.

©2013-2026 Tencent Cloud. All rights reserved.

2. Select the mount type **Resource Group Cache**, set the File Type to **Model**, and select the file **model_file** from the File Name dropdown list.
3. Submit the task to mount the preloaded model file "model_file" from the selected resource group A to the dev machine.



©2013-2026 Tencent Cloud. All rights reserved.

Fault Diagnosis and Recovery

Last updated: 2026-02-26 16:03:40

Overview

The feature of fault diagnosis and recovery aims to timely detect exceptions in computing resources, quickly identify the issue type and root cause, and restore computing resources through automatic or manual methods, ensuring business continuity and stability.
This document provides a detailed introduction to the mechanism for fault diagnosis provided by Tencent Cloud TI-ONE Platform (TI-ONE) and lists typical exception scenarios along with their handling solutions. It also introduces how to configure alarm rules to provide timely warnings and minimize business risks to the greatest extent.

Mechanism for Fault Diagnosis

The platform provides both automatic and manual diagnostic methods to comprehensively cover various Ops scenarios.

1. **Automatic diagnosis**
   - **CVM automatic diagnosis**: Cloud Virtual Machine (CVM) supports automatic detection of CVM instance exceptions and proactively issuing maintenance tasks.
   - **TI-ONE automatic detection**: TI-ONE supports regular detection of whether nodes are available in the background to prevent potential failures.
2. **Manual diagnosis**
   - **TI-ONE health check**: In TI-ONE's resource groups, you can manually create health check tasks to proactively check the network connectivity and environment consistency of nodes.

CVM Automatic Diagnosis

CVM automatic diagnosis is a standardized troubleshooting service provided by CVM. When CVM detects a sudden instance exception (such as a sudden crash of the underlying host, or predicting potential software or hardware failures in the underlying host to avoid a crash), it automatically creates corresponding maintenance tasks and sends notifications. You can view and track instance recovery status in the maintenance task list in the CVM console. For types of exceptions that trigger maintenance tasks, their specific meanings, and handling policies, see Maintenance Task Types and Handling Policies.

> **Note:**
> If a CVM instance you added to TI-ONE resource group has a maintenance task issued by CVM, the platform will change the node status to Pending Maintenance and guide you

©2013-2026 Tencent Cloud. All rights reserved.

to the CVM console to complete authorization operations. For handling methods, see CVM Maintenance Tasks on Nodes.

TI-ONE Automatic Diagnosis

TI-ONE automatic diagnosis is a fault diagnosis capability provided by the TI-ONE resource group. It continuously monitors node status and checks whether associated components are available during node addition, management, and release processes. When it detects issues such as GPU card disconnections, XID exceptions, or Virtual Private Cloud (VPC) network unreachability, it can pinpoint specific causes using diagnostic tools and automatically classify them into three handling methods.

Handling Method	Description	Example
Handled by TI-ONE in a closed loop	TI-ONE completes the entire process of diagnosis, troubleshooting, and recovery. The console only displays necessary information (including the exception cause and exception messages).	When a CVM node is added to the resource group, the operation failed due to the failure to create a Tencent Kubernetes Engine (TKE) cluster.
Jointly handled by TI-ONE & users	After TI-ONE detects an exception, it notifies the user to intervene. After guiding the user to complete certain operations (such as authorization), TI-ONE resolves the issue.	A node already added to a resource group has a maintenance task issued by CVM. User authorization is required before TI-ONE & CVM can recover the node.
Handled by users	TI-ONE provides detected exception information to users and guides them to perform troubleshooting operations independently.	The computing resource fee for a CVM instance or the software subscription fee for a node is about to expire, and users need to renew it manually.

TI-ONE Health Check

©2013-2026 Tencent Cloud. All rights reserved.

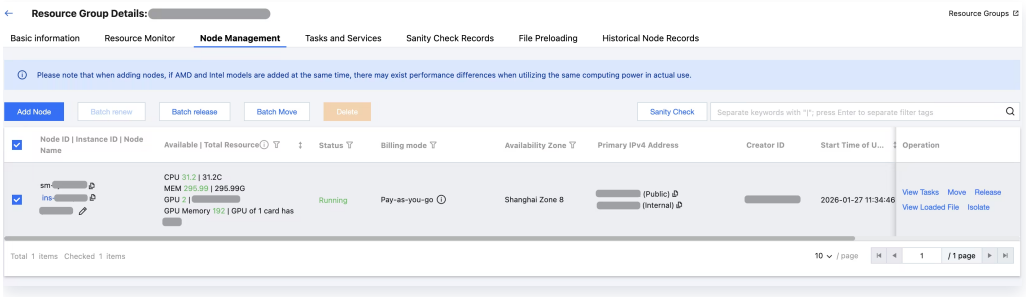
TI-ONE health check is a proactive fault diagnosis capability provided by TI-ONE resource groups. It allows you to proactively initiate health check tasks after adding nodes to verify key parameters such as network communication performance and environment consistency, ensuring that operational requirements are met.
By pre-detecting node availability, it effectively avoids various operational issues caused by node failures, for example:

- Avoiding GPU resource waste: When tasks or services fall into a continuous retry loop due to node failures (such as driver exceptions), they cannot execute normally, even after time-consuming initialization operations like model loading are completed. This not only occupies valuable GPU resources but also requires additional time for troubleshooting and task resubmission. The health check can identify and exclude faulty nodes before such tasks start, ensuring efficient resource utilization.
- Preventing performance degradation of tasks/services: During the execution of tasks or services, issues like slow network communication in nodes may lead to inefficient training or delayed responses in online services. Such performance bottlenecks are difficult to predict before a task starts, and are often detected only after causing substantial impact. The health check can detect these potential performance flaws in advance, preventing tasks from starting on unhealthy nodes and safeguarding business performance from the source.

The following provides a detailed introduction to how to create a health check task within a resource group.

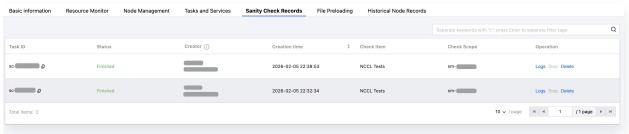
1. **Creating a health check task**

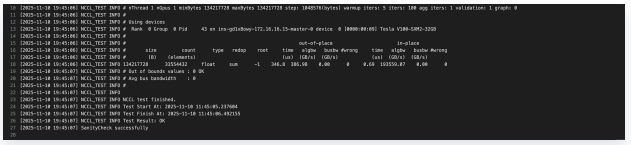
Click the **Node Management** tab on the resource group details page, select the range of nodes to be checked, and click **Health Check** in the upper right corner. Select the check items in the pop-up window, and the creation is completed.



2. **Viewing task records and results**

Switch to the **Health Check Records** tab to view task execution history and logs of check results.





Typical Scenarios and Handling Solutions

©2013-2026 Tencent Cloud. All rights reserved.

Handled by TI-ONE in a Closed Loop

The table below lists over a dozen exception scenarios that TI-ONE automatically diagnoses and repairs. When the platform detects such exceptions, it will automatically trigger the handling process and coordinate relevant resources for repair, requiring no manual intervention. Please wait patiently. If the issue persists, you can submit a ticket to contact us for further support.

Triggering Stage	Node Status	Exception Cause	Exception Scenario
Adding nodes	Deployment failed	Failure to create a TKE cluster	• Cluster creation by calling the TKE API fails due to TKE version restrictions. • Adding an agent node to the cluster fails due to unsynchronized changes in the TKE API. • Node registration of the TKE cluster fails because its node registration feature is not enabled or contains a bug.
		Failure to add nodes to the TKE cluster	• Execution of the registration script fails due to bugs in the registration script or incompatibility with the new operating system version. • OS installation by calling the CVM API fails because the original OS image was no longer supported due to CVM operating system changes. • Execution of the registration script fails because the hard disks of CVM nodes have read-only permissions.
		Failure to initialize the TKE cluster	• System components of the TKE cluster are in the pending status due to GPU/RDMA failures on the CVM node. • System components of the TKE cluster are in a crash status because network exceptions on the CVM node prevent access to system services. • Resources like qGPU, RDMA, and ENI show as 0 because TKE is not compatible with the GPU model of the CVM node. • The status of the Turbo CFS component is abnormal. The storage component on the CVM node is unavailable because Turbo CFS did not adapt to the latest operating system.
Using nodes	Abnormal	Node not in the	• The node is unreachable via ping or SSH because the CVM node failed to start or was manually stopped.

©2013-2026 Tencent Cloud. All rights reserved.

		Running status	• Node resources are abnormal (for example, qGPU/RDMA/ENI resources show as 0) because critical service components failed to initialize after the CVM node started.
		Exceptions detected on the node by Node Problem Detector (NPD)	• The CVM node is unavailable due to exceptions in objects such as the GPU, RDMA, CPU, memory, local disk, operating system, or Kubernetes components.
		Abnormal system components of the TKE cluster	• The RDMA/GPU components of the CVM node are in a crash status due to the failure of the RDMA NIC. • The CVM node is unavailable due to a memory leak in the operating system.
		Abnormal qGPU component status	• The qGPU component on the CVM node is unavailable because qGPU did not adapt to the corresponding GPU model, or the driver version is later than 550.

For the above scenarios, the TI-ONE console will display necessary prompts, including the failure cause and specific exception details.

Jointly Handled by TI-ONE & Users

The table below lists 4 exception scenarios that require collaborative handling by TI-ONE and users. When such exceptions are detected, the platform will clearly indicate the cause and guide you to the designated page to complete necessary troubleshooting or repair operations.

Triggering Stage	Node Status	Exception Cause	Exception Scenario

©2013-2026 Tencent Cloud. All rights reserved.

Adding nodes	Deployment failed	Failure to add nodes to the TKE cluster	• The node cannot be connected due to network restrictions in the user's VPC network.
			• Execution of the registration script fails due to residual LVM volumes on the CVM node.
Using nodes	Pending maintenance	Maintenance tasks issued by CVM on nodes	• CVM proactively detects node exceptions and issues a maintenance task (hardware failures such as CPU/hard disk/main board/NIC, or operational exceptions such as GPU/NIC). For details, see CVM Automatic Diagnosis
	Abnormal	Abnormal network connection	• The node cannot be connected due to abnormal configurations of the security group/route in the user's VPC network.

Handling solutions for various exceptions:

Deployment Failure or Connection Exception Caused by VPC Network Restrictions

Solution:

1. Click **CCN** in the prompt to go to the CCN console.
2. **Create instance**: Click **New** to create a CCN instance. For details, see Creating a CCN Instance.
3. **Associate VPC**: Go to the CCN instance details page. In the **Associate Instance** tab, click **Add Instance** at the top left of the list. In the pop-up window, select the network instance type, region, and specific VPC instance to be associated. For details, see Associating a Network Instance. (Remarks: The VPC instances to be added are the Platform VPC and Node VPC displayed on the console.)
4. **Check route table**: In the Route Table tab, check whether the routing policies of each subnet under VPC associated with CCN have taken effect. If the IP ranges of the associated network instances conflict, invalid routes will be generated.

CVM Maintenance Tasks on the Node

Solution:

1. Click **Authorize** in the prompt to go to the list of Maintenance Tasks in the CVM console and find the task information on the corresponding CVM instance.
2. Click **Authorize/Schedule** on the right side of the list. In the pop-up window, select the specific authorization maintenance method and schedule a maintenance time. Click

©2013-2026 Tencent Cloud. All rights reserved.

Confirm to complete the authorization operation.

Residual LVM Volumes on the Node

Solution:

1. Click **CVM console** in the prompt to go to the CVM console and find the corresponding CVM instance in the instance list.
2. Go to the instance details page. In the list of data disks mounted to the instance, click **Unmount** to complete the cleanup.

Handled by Users

The table below lists 3 common scenarios that users can handle completely on their own. After detecting these exceptions, the platform will provide clear exception descriptions and handling instructions. You can follow the prompts to independently complete the repair operations on the specified page and quickly restore business operations.

Triggering Stage	Node Status	Exception Cause	Exception Scenario
Using nodes	Abnormal	Node not in the Running status	• The node is unavailable because the user manually restarted/shut down the node in the CVM console, or the node has expired.
	Running	Node about to expire	• The computing resource fee for the CVM instance is about to expire and requires renewal by users. • The software subscription fee for the TI-ONE resource group node is about to expire and requires renewal by users.

Handling solutions for various exceptions:

Node not in the Running status

Solution:

1. Click **CVM console** in the prompt to go to the CVM console and find the corresponding CVM instance in the instance list.
2. Confirm whether the instance status is Restarting/Shut Down/Pending Recycling.

©2013-2026 Tencent Cloud. All rights reserved.

2.1 If the CVM instance is in the above statuses and you confirm that it is no longer needed, you can go to Instance Recycle Bin and click **Release**.
2.2 If you still need to use the instance, you can click **Renew** or **Restore** to ensure it remains available.

Node about to expire

- **Solution 1: CVM computing resource fee about to expire**

1. Click **CVM console** in the prompt to go to the CVM console and find the corresponding CVM instance in the instance list.
2. Click **Renew** on the right, select Duration for Renewal, and then click **Confirm**. For details, see Renewing Instances.

- **Solution 2: TI-ONE software subscription fee about to expire**

Click **Renew** on the right side of the list, select Duration for Renewal, and then click **Confirm**.

Alarm Configuration

During the Ops process of resource groups, the platform has integrated monitoring of node status into the alarm rules of Tencent Cloud Observability Platform (TCOP). When nodes enter critical lifecycle statuses such as Abnormal, Under Maintenance, or Isolated, the system will proactively send alarms to you. This helps you stay informed about resource dynamics and ensures Ops efficiency.

- The currently covered alarm-triggering statuses include Abnormal, Purchasing, Under Maintenance, Pending Maintenance, Running, Deleted, Isolated, and more.

The steps to configure alarm rules are as follows:

1. Choose TCOP > Alarm Management > Alarm Configuration, click **New Policy**, and select **Custom Alarm Policy**.
2. On the Create New Alarm Policy page, fill in the policy name and description, and configure alarm rules according to the following parameters:
   2.1 Monitoring type: Select **Cloud Product Monitoring**.
   2.2 Policy type: Select **Tencent Cloud TI-ONE Platform / Resource / Resource Status**.
   2.3 Alarm object: Select Specified Instances (nodes of the resource group) or All Objects as needed.

©2013-2026 Tencent Cloud. All rights reserved.

2.4 Trigger condition: Configure manually. In the Metric field, select the node status that requires an alarm, such as Instance Abnormal.

3. Click **Next: Configure Alarm Notification**, select a notification template, and then click **Complete**.

©2013-2026 Tencent Cloud. All rights reserved.

Resource Monitor
Introduction to Resource Monitor

Last updated: 2026-02-26 16:01:50

Resource Monitor provides resource dashboards for you to view metrics related to resources and business, allowing you to evaluate resource usage and task operation status. Currently, you can view the Resource Monitor panel in Enterprise Management and Workspace. Within the Workspace, you can only view metrics related to the current workspace's business, while Enterprise Management allows you to view resource and business-related metrics across all workspaces.

- **Resource dimension:**
  - Supports displaying monitoring metrics by a resource dimension, including the number of GPU cards, GPU allocation rate, task utilization, and GPU utilization.
  - Supports viewing monitoring metrics of resource groups by a dimension of resource groups, including the number of GPU cards, GPU allocation rate, task utilization, and GPU utilization.
  - Supports displaying the GPU card hours occupied by each GPU model, categorized by usage team tag.
- **Business dimension:**
  - Supports displaying monitoring metrics by a business/task dimension, including the number of tasks and services, tasks that have been run, and low-utilization tasks.
  - Supports displaying metrics such as GPU utilization and first token latency by a service dimension (only for services with monitoring of LLMs).

©2013-2026 Tencent Cloud. All rights reserved.

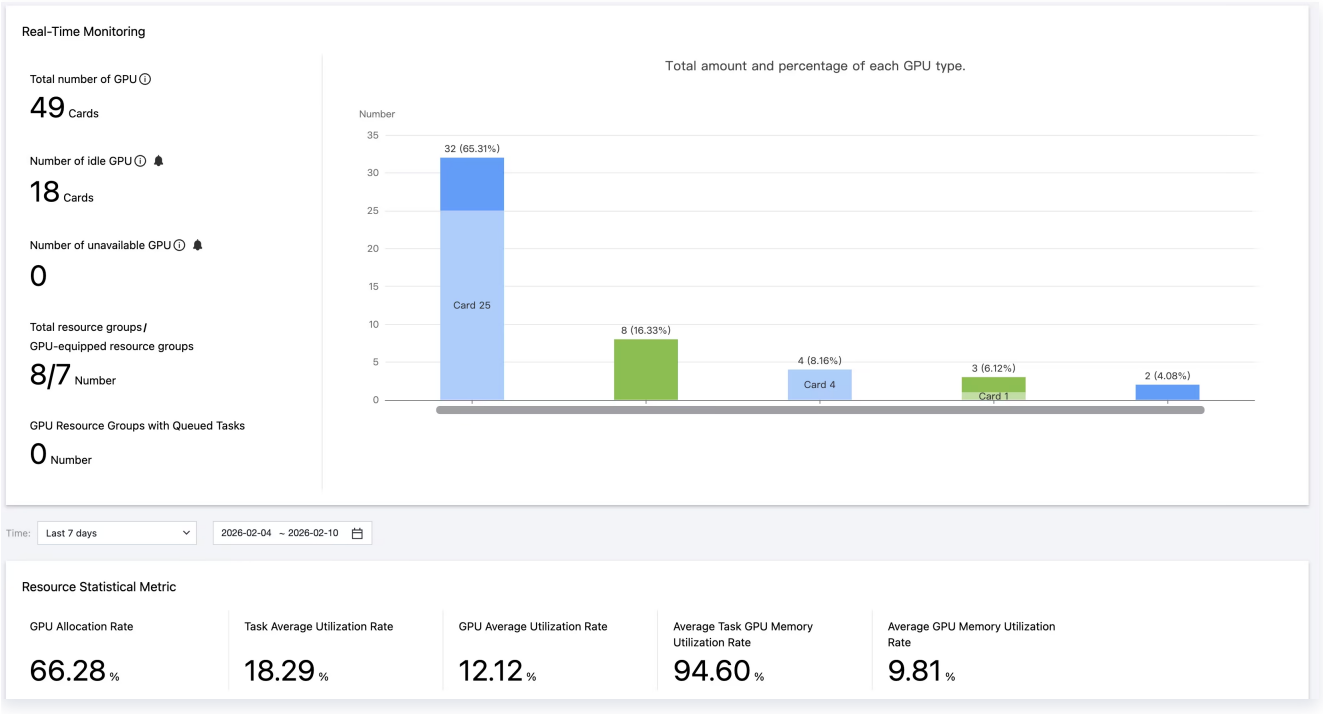
Metric Description

Last updated: 2026-02-26 16:02:01

Resource Monitor provides resource dashboards for you to view metrics related to resources and business, allowing you to evaluate resource usage and task operation status.

You can select one or more resource groups, and the system will aggregate and display the resource usage of these resource groups. Key metrics include:

- GPU allocation rate: measures the proportion of GPU card hours used by tasks to the total available GPU card hours, reflecting the level of demand saturation for resources.
- Average GPU utilization: evaluates the percentage of time that the allocated GPU card is actually in a computational state, reflecting the true efficiency of resource usage.
- Average task utilization: evaluates the actual computational busyness level of tasks during resource occupation.

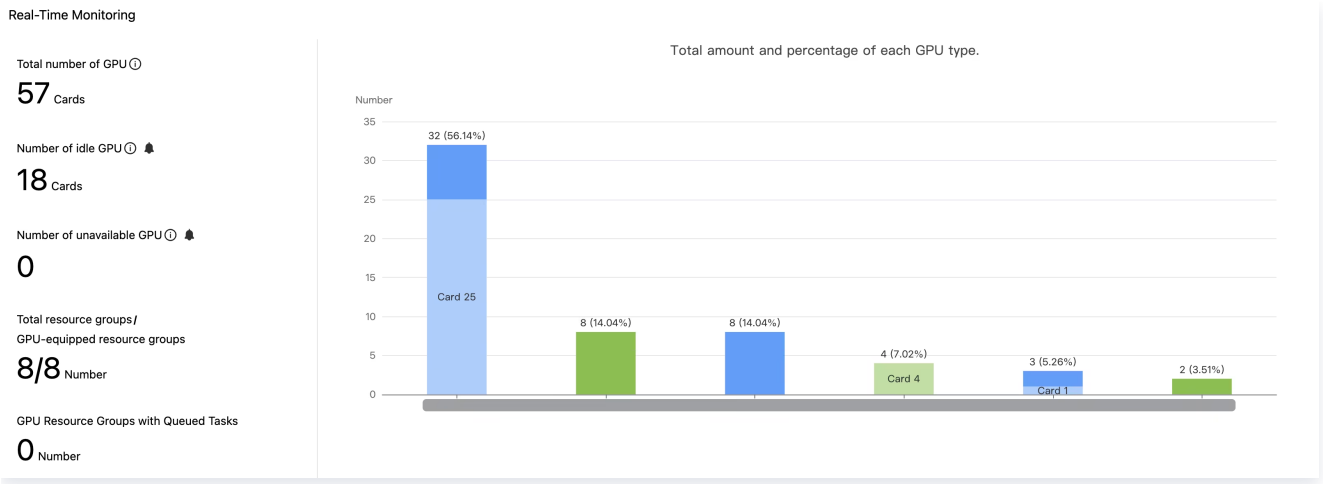


1. Resource Dimension

Detailed Metric Description

Real-Time Monitoring

©2013-2026 Tencent Cloud. All rights reserved.



Metric Name	Description
Total number of GPU cards	Total number of GPU cards within the current resource group, including yearly/monthly subscription GPU cards, pay-as-you-go GPU cards, and GPU cards from nodes deployed in the resource group.
Number of idle available GPU cards	Number of GPU cards currently unused and available
Number of unavailable GPU cards	Number of currently unused and unavailable GPU cards
Total resource groups/GPU-included resource groups	Total resource groups include all resource groups visible based on user permissions.
Proportion of each GPU model	The current number of GPU cards for each GPU model and its proportion to the total number of GPU cards. The histogram simultaneously displays the number of occupied GPU cards.

Notes on whether to include the total number of GPU cards, number of idle available GPU cards, and number of unavailable GPU cards based on the status of resource group nodes:

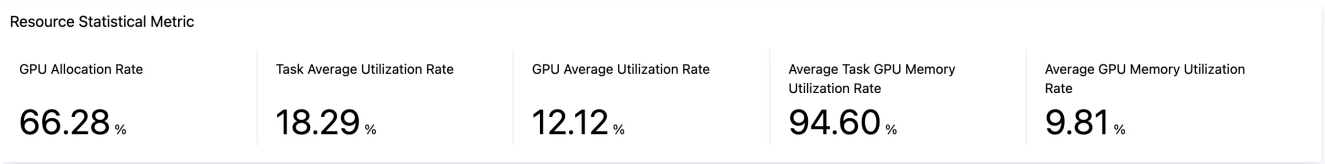
Status of Resource Group Node	Total Number of GPU Cards	Number of Idle Available GPU Cards	Number of Unavailable GPU Cards
Purchasing	No	Not included	Not included

©2013-2026 Tencent Cloud. All rights reserved.

Deploying	No	Not included	Not included
Running	Yes	Included	Not included
Mounting disk	Yes	Included	Not included
Node moving	Yes	Not included	Included
Abnormal	No	Not included	Included
Node under maintenance	Yes	Not included	Included
Expired due to overdue payment	No	Not included	Included
Pending maintenance	Yes	Not included	Included
Isolated	Yes	Not included	Included
CVM resource expired	No	Not included	Included
Deployment failed	No	Not included	Not included
Releasing	No	Not included	Included
Released	No	Not included	Included
Managed and no longer used	No	Not included	Included

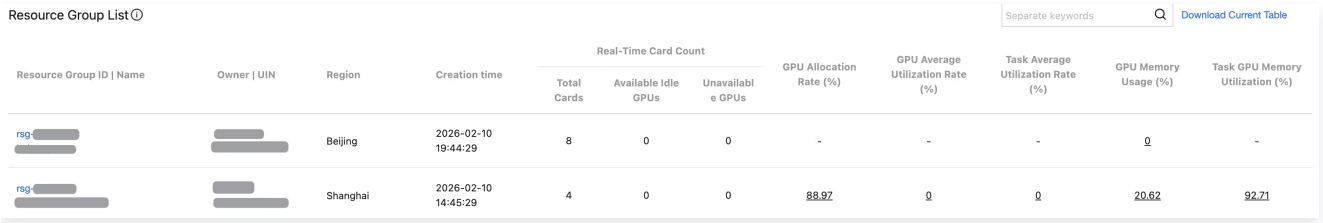
Statistical Metrics

The system supports periodic statistics on metrics (test phase data is retained for reference; valid data is counted starting from November 1). Statistical metrics include: weighted GPU allocation rate, average task utilization, and average GPU utilization across all Pods in selected resource groups. The resource group list displays GPU allocation rate, average task utilization, and average GPU utilization for each resource group. The GPU card hour consumption list shows statistics on the occupied GPU card hours for each resource group.
Resource statistical metrics:



©2013-2026 Tencent Cloud. All rights reserved.

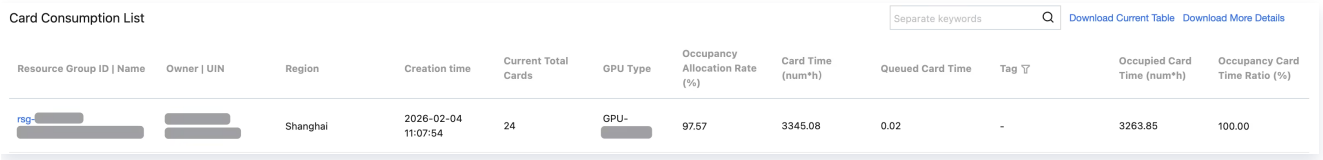
Resource group list:


Resource Group List ⓘ

Separate keywords | Download Current Table

Resource Group ID | Name	Owner | UIN	Region	Creation time	Real-Time Card Count			GPU Allocation Rate (%)	GPU Average Utilization Rate (%)	Task Average Utilization Rate (%)	GPU Memory Usage (%)	Task GPU Memory Utilization (%)
				Total Cards	Available Idle GPUs	Unavailable GPUs					
rsg-		Beijing	2026-02-10 19:44:29	8	0	0	-	-	-	0	-
rsg-		Shanghai	2026-02-10 14:45:29	4	0	0	88.97	0	0	20.62	92.71

GPU card-hours consumption list:


Card Consumption List

Separate keywords | Download Current Table | Download More Details

Resource Group ID | Name	Owner | UIN	Region	Creation time	Current Total Cards	GPU Type	Occupancy Allocation Rate (%)	Card Time (num*h)	Queued Card Time	Tag	Occupied Card Time (num*h)	Occupancy Card Time Ratio (%)
rsg-		Shanghai	2026-02-04 11:07:54	24	GPU-	97.57	3345.08	0.02	-	3263.85	100.00

Metric Name	Description
GPU allocation rate	Proportion of allocated and used GPU card hours to total GPU card hours within a given time range. For specific calculation formulas, see Task Utilization/GPU Utilization/Allocation Rate of Resource Groups in the Core Metric Calculation Description section (health monitoring tasks are currently excluded from statistics).
Average task utilization	Weighted average (by GPU card hours) of single-instance actual GPU utilization during the statistical period. For specific calculation formulas, see Task Utilization/GPU Utilization/Allocation Rate of Resource Groups in the Core Metric Calculation Description section (health monitoring tasks are currently excluded from statistics).
Average GPU utilization	Weighted sum (by GPU card hours) of single-instance actual GPU utilization divided by the total GPU card hours during the statistical period. For specific calculation formulas, see Task Utilization/GPU Utilization/Allocation Rate of Resource Groups in the Core Metric Calculation Description section (health monitoring tasks are currently excluded from statistics).
Occupied allocation rate	Proportion of occupied GPU card hours to total GPU card hours within a given time range. For specific calculation formulas, see Task Utilization/GPU Utilization/Allocation Rate of Resource Groups in the Core Metric Calculation Description section (health monitoring tasks are currently excluded from statistics).
Occupied GPU card hours	GPU card hours occupied within a given time range.

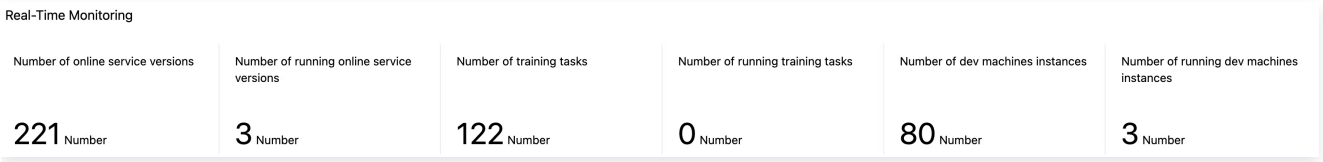
Where the resources of a resource group are used by different team members, the GPU card-hours consumption list supports statistics by team member. Users need to tag tasks with the key **ti_sys_usergroupid** when creating tasks.

©2013-2026 Tencent Cloud. All rights reserved.

2. Business Dimension

Detailed Metric Description

Real-Time Monitoring

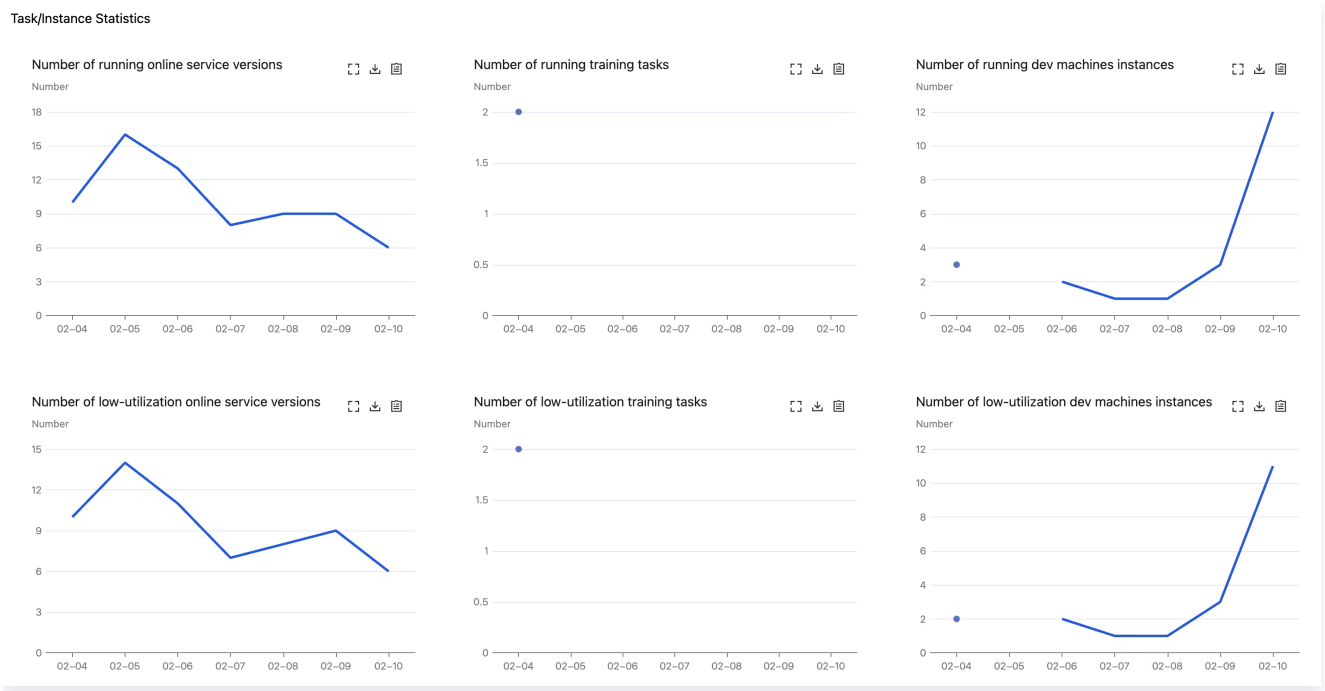


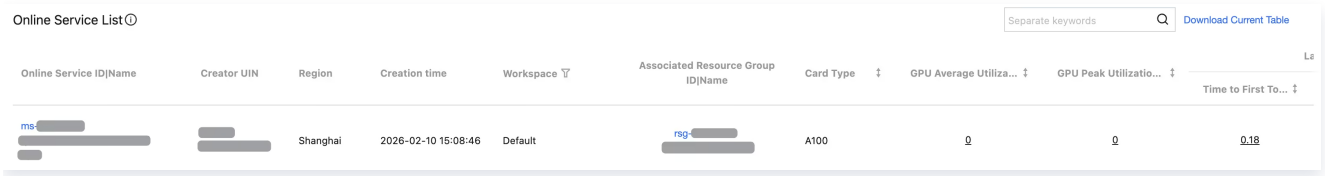
Metric Name	Description
Number of online services versions	Number of versions for all Online Services instances on the current platform
Number of running online services versions	Number of versions of running Online Services instances on the current platform
Number of training tasks	Total number of training tasks on the current platform
Number of running training tasks	Total number of training tasks running on the current platform
Number of dev machines	Number of dev machines on the current platform
Number of running dev machines	Number of dev machines running on the current platform

Statistical Metrics

The system supports periodic statistics on metrics (test phase data is retained for reference; valid data is counted starting from November 1). Statistical metrics include the number of tasks/services that have been run and low-utilization tasks/services. Each LLM service displays metrics such as GPU allocation rate, average GPU utilization, latency, and number of requests per minute.

©2013-2026 Tencent Cloud. All rights reserved.





Metric Name	Description
Number of online services versions that have been run	Trend chart of the number of service instance versions that reported GPU usage during the statistical period.
Number of training tasks that have been run	Trend chart of the total number of training tasks that reported GPU usage during the statistical period.
Number of dev machines that have been run	Trend chart of the number of dev machines that reported GPU usage during the statistical period.
Number of low-utilization service versions	Trend chart of the number of tasks (with single-task GPU utilization below 50%) that have been run during the statistical period.
Number of low-utilization training tasks	Trend chart of the number of tasks (with single-task GPU utilization below 50%) that have been run during the statistical period.
Number of low-utilization dev machines	Trend chart of the number of tasks (with single-task GPU utilization below 50%) that have been run during the statistical period.

©2013-2026 Tencent Cloud. All rights reserved.

Average GPU utilization	Weighted sum (by GPU card hours) of single-instance actual GPU utilization divided by the total GPU card hours during the statistical period. For specific calculation formulas, see Task Utilization/GPU Utilization/Allocation Rate of Resource Groups in the Core Metric Calculation Description section (health monitoring tasks are currently excluded from statistics).
Peak GPU utilization	Weighted sum (by GPU card hours) of the maximum single-instance actual GPU utilization divided by the total GPU card hours during the statistical period. For specific calculation formulas, see Task Utilization/GPU Utilization/Allocation Rate of Resource Groups in the Core Metric Calculation Description section (health monitoring tasks are currently excluded from statistics).
Other metrics (such as latency and number of requests per minute)	For details, see the Description of Service Monitoring section in Operations of Online Services.

3. Core Metric Calculation Description

Calculation Process of the Average/Peak Utilization of GPU Tasks

A GPU task consists of multiple Pods, and each Pod contains multiple GPU cards. The minimum granularity for utilization reporting is per GPU card, with each GPU card reporting a utilization value every minute.
The average/peak utilization of a single task is aggregated from the utilization of all GPU cards used by the task during its lifecycle.

Note:

- For each task, the average utilization and peak utilization are calculated every day. (For tasks running for multiple days, they are sliced by day to calculate average/peak utilization. For example, a task running for 3 days will be sliced into 3 separate tasks for calculation.)
- The average utilization and peak utilization of each task are calculated by first calculating the GPU utilization for each Pod, then weighting the results of different Pods by GPU card hours.
- The platform allows different Pods within the same task to use different GPU models. When aggregating by GPU models, the calculation is weighted by GPU card hours based on the Pods using that specific GPU model.

©2013-2026 Tencent Cloud. All rights reserved.

Task Utilization/GPU Utilization/Allocation Rate of Resource Groups

- Pod GPU card hours (T): T = number of GPU cards used by the Pod x running duration (in hours), T_i represents the GPU hours of the i-th Pod.
- Average task utilization (Ravg): Calculation process as described in Figure 1. $Ravg_i$ represents the average utilization of the i-th task (Pod).
- Number of GPU cards of resource group (Q): Q_t represents the number of GPU cards in the resource group t.
- Task utilization of resource group = $\frac{\sum_{i=1}^{n} Ravg_i * T_i}{\sum_{i=1}^{n} T_i}$, where n represents the number of Pods after slicing by day.
- Allocation rate of resource group = $\frac{\sum_{i=1}^{n} T_i}{\sum_{t=1}^{7} Q_t * 24}$, where 7 represents a 7-day statistical period, and 24 indicates that GPU cards in the resource group t persist throughout the 7 days with a daily calculation duration of 24 hours. In practice, if a GPU card operates for less than 24 hours, the actual duration is used. GPU card hours for the allocation rate start counting from when the main container begins running. For the occupied allocation rate, GPU card hours start counting from when the Pod completes scheduling.
- GPU utilization of resource group = $\frac{\sum_{i=1}^{n} Ravg_i * T_i}{\sum_{t=1}^{7} Q_t * 24}$, where 7 represents a 7-day statistical period, and 24 indicates that GPU cards in the resource group t persist throughout the 7 days with a daily calculation duration of 24 hours. In practice, if a GPU card operates for less than 24 hours, the actual duration is used

©2013-2026 Tencent Cloud. All rights reserved.

GPU Virtualization

Last updated: 2026-02-26 16:05:00

Overview

The GPU virtualization feature provided by Tencent Cloud TI-ONE Platform (TI-ONE) allocates the computing power of a single GPU card to different training tasks and inference services, enhancing resource allocation flexibility and utilization efficiency.

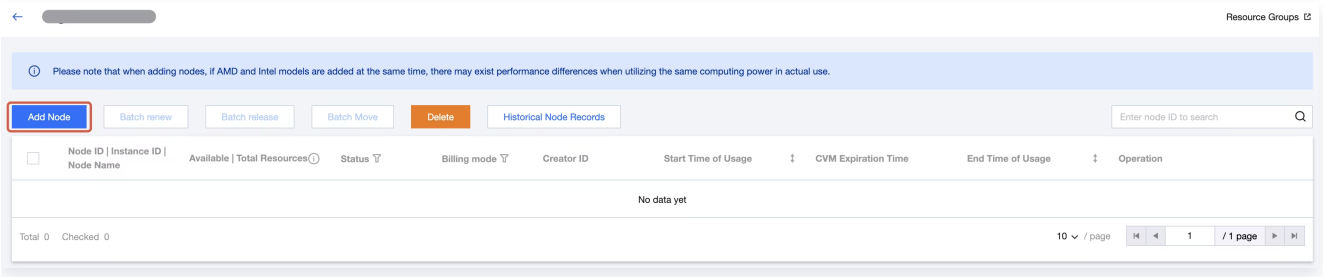
Supported GPU Models

GPU models supported by the GPU virtualization feature of the platform are as follows:
T4, V100, A100, A10, PNV5b, A800, HCCA100, HCCPNV6, and PNV6.

Feature Use Instructions

Prerequisites

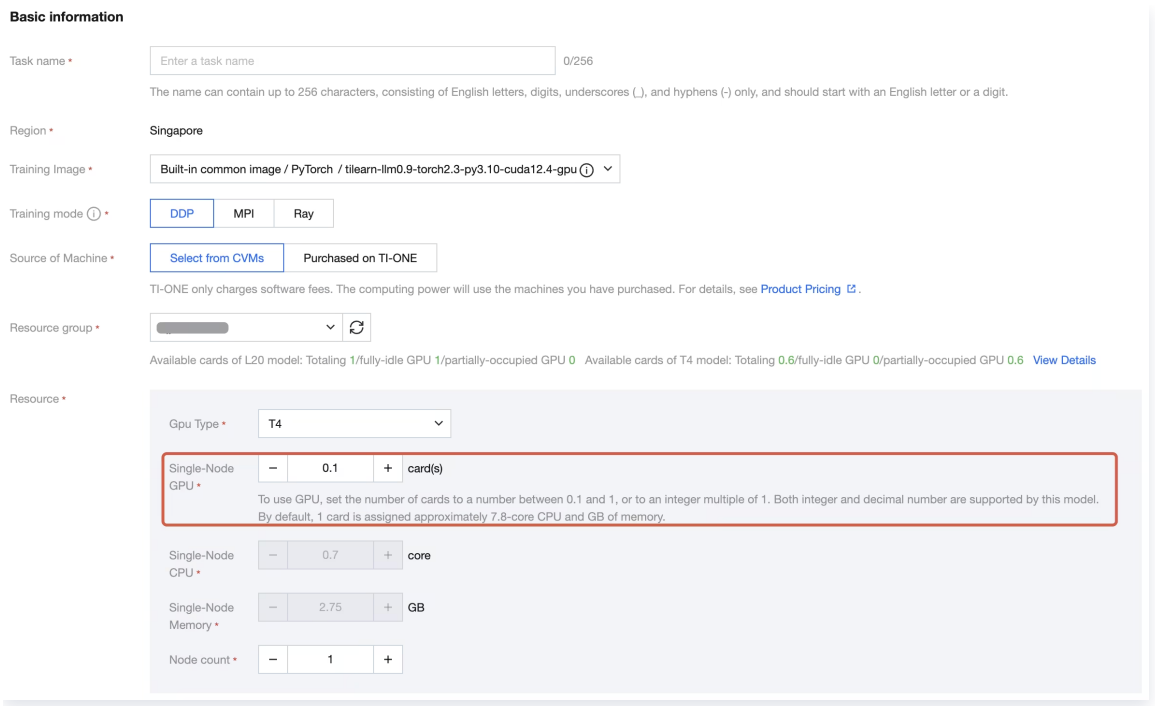
You have created a resource group on the Resource Group Management page and added your Cloud Virtual Machine (CVM) instance to the resource group as a node.



Usage Method

During the creation of a dev machine, a task-based modeling training task, or an online service, if your resource group includes GPU card models that support virtualization, you can set the number of GPU cards to a value ranging from 0.1 to 1 when configuring resources.

©2013-2026 Tencent Cloud. All rights reserved.



Must-Knows

As the number of GPU card models on the market continues to increase and related drivers are constantly being updated, the GPU virtualization feature of the platform gradually supports more GPU models than before. Newly supported virtualized GPUs may not be able to use the virtualization feature in existing resource groups. Pay attention to the relevant prompts of the platform when using the GPU.

- If a node with a GPU card model that the platform does not yet support for virtualization is added when a resource group is created, the virtualization feature of the GPU in the resource group still cannot be used after the platform subsequently supports virtualization for that GPU card model.
- In this case, you cannot directly move the node to another resource group, but you can remove it from the resource group and re-add it to the new resource group. In this way, the virtualization feature can be used.

©2013-2026 Tencent Cloud. All rights reserved.

Tikit
Tikit Introduction

Last updated：2026-02-26 15:36:11

Background

The platform provides the Python development kit (Tikit) to help algorithm engineers better interact with the platform in Notebook or integrated development environments (IDEs). This document will introduce how to use this kit.

©2013-2026 Tencent Cloud. All rights reserved.

Installation and Initialization

Last updated: 2026-02-26 15:36:53

Operation Steps

Environment Dependencies

Currently, Python 3.4 and later versions are supported.

Installing Tikit

Tikit has been built in containers for Notebook and training tasks on Tencent Cloud TI-ONE Platform (TI-ONE), so you do not need to install Tikit.
In a non-public-cloud TI-ONE environment, the method for installing Tikit is as follows:

- Install dependencies:

```
#centos:

sudo yum -y install cyrus-sasl cyrus-sasl-devel cyrus-sasl-lib

#ubuntu:

sudo apt-get update
sudo apt-get install -y libsasl2-dev
```

- Method 1: Install Tikit using pip (recommended).

```
pip3 install -U tikit
```

- Method 2: Install Tikit using an offline method. You can download the installation package from https://pypi.org/project/tikit/ , and then use the whl file or source code to install Tikit:

```
pip3 install tikit-1.0.0-py3-none-any.whl

# Alternatively, install Tikit after the source code is decompressed:
python3 setup.py install
```

Using Tikit

1. Prepare the secret_id and secret_key information.

©2013-2026 Tencent Cloud. All rights reserved.

Log in to Tencent Cloud and obtain them from the page as follows:

2. Complete the initialization operation and call the function.

```
from tikit.client import Client

# Initialize the client. In the environment of the public cloud TI-ONE
(including Notebook and training tasks), you do not need to enter the
region information because it is already available in the environment
variables.
client = Client("your_secret_id", "your_secret_key", "<region>")
```

©2013-2026 Tencent Cloud. All rights reserved.

Viewing Resources in a Resource Group

Last updated: 2026-02-26 16:06:02

```
describe_resource_groups(self, offset=0, limit=20, search_word="",
tag_filters=None):
    Obtain the resource group list.
        :param offset: offset. The default value is 0. For the starting
position of a pagination query, if Limit is 100, the offset for the first
page is 0, the offset for the second page is 100, and so on. That is, the
left side of each page is an open interval.
        :type offset: int
        :param limit: number of returned results.
        :type limit: int
        :param search_word: You can perform fuzzy searches for resource
group IDs and resource group names.
        :type search_word: str
        :param tag_filters: tag filters.
        :type tag_filters: list of
tikit.tencentcloud.tione.v20211111.models.TagFilter
        :return:
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeBillingResourceG
roupsResponse`
```

©2013-2026 Tencent Cloud. All rights reserved.

Uploading and Downloading Data

Last updated: 2026-03-10 10:16:57

1. Directly Called Functions

1.1. API List

```
download_from_hdfs(hdfs_url, hdfs_path, local_path)
    Download files from HDFS to a local machine.

    :param hdfs_url:    WebHDFS address, for example,
http://10.0.3.16:4008.
    :type hdfs_url:     str
    :param hdfs_path:   path on HDFS.
    :type hdfs_path:    str
    :param local_path:  local path.
    :type local_path:   str
    :return:            local result path.
    :rtype:             str

upload_to_hdfs(local_path, hdfs_url, hdfs_path)
    Upload a local directory to HDFS.

    :param local_path:  local path.
    :type local_path:   str
    :param hdfs_url:    WebHDFS address, for example,
http://10.0.3.16:4008.
    :type hdfs_url:     str
    :param hdfs_path:   path on HDFS.
    :type hdfs_path:    str
    :return:            result path of HDFS.
    :rtype:             str

upload_to_hive_by_hdfs(local_path, hdfs_url, hive_server, table_name,
database='default', auth='CUSTOM', username=None, password=None,
overwrite=False, partition='')
    Import the data in a local file into a Hive table with HDFS as the
intermediate storage.
    Process: Upload the local file to HDFS first, and then import the file
from HDFS into the Hive table.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param local_path:      local file or folder. The folder cannot
contain subfolders.
    :type local_path:       str
    :param hdfs_url:        WebHDFS URL, for example,
http://10.0.3.16:4008.
    :type hdfs_url:         str
    :param hive_server:     HiveServer2 address.
    :type hive_server:      str
    :param table_name:      Hive table name.
    :type table_name:       str
    :param database:        database name.
    :type database:         str
    :param auth:            authentication method.
    :type auth:             str
    :param username:        username for database authentication.
    :type username:         str
    :param password:        password for database authentication.
    :type password:         str
    :param overwrite:       Whether to delete the original data.
    :type overwrite:        bool
    :param partition:       partition selection.
    :type partition:        str
    :return:
    :rtype:

export_from_hive_by_hdfs(local_path, hdfs_url, hive_server, table_name='',
sql='', database='default', auth='CUSTOM', username=None, password=None,
row_format="row format delimited fields terminated by ','")
    Export a Hive table to a local machine with HDFS as the intermediate
storage. For large files, this method is more efficient than that of
directly writing from the Hive table to the local machine.
    Process: Export the Hive table to HDFS first, and then download it
from HDFS to the local machine.

    :param local_path:      local directory.
    :type local_path:       str
    :param hdfs_url:        WebHDFS URL, for example,
http://10.0.3.16:4008.
    :type hdfs_url:         str
    :param hive_server:     HiveServer2 address.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :type hive_server:      str
    :param table_name:      Hive table name. This parameter can be ignored
when SQL is specified.
    :type table_name:       str
    :param sql:             SQL statement for querying data, for example,
select * from t1.
    :type sql:              str
    :param database:        database name.
    :type database:         str
    :param auth:            authentication method.
    :type auth:             str
    :param username:        username for database authentication.
    :type username:         str
    :param password:        password for database authentication.
    :type password:         str
    :param row_format:      row output format.
    :type row_format:       str
    :return:
    :rtype:
```

1.2. Usage:

```
import tikit

tikit.upload_to_hdfs("dir1/file1", "http://10.0.3.16:4008", "/dir1/file1")
```

2. Methods Called Through the Client

```
describe_cos_buckets(self)
    List all buckets.

    :return:    bucket list.
    :rtype:     dict
    The returned results are as follows:
    {
      "Owner": {
        "ID": "qcs::cam::uin/100011011262:uin/100011011262",
        "DisplayName": "100011011162"
      },
      "Buckets": {
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        "Bucket": [
          {
            "Name": "bucket-58565",
            "Location": "ap-beijing-fsi",
            "CreationDate": "2021-07-21T11:06:00Z",
            "BucketType": "cos"
          },
          {
            "Name": "tai-1300158565",
            "Location": "ap-guangzhou",
            "CreationDate": "2021-10-22T11:04:40Z",
            "BucketType": "cos"
          }
        ]
      }
    }

describe_cos_path(self, bucket, path, maker='', max_keys=1000,
encoding_type='')
    Obtain the information on the Cloud Object Storage (COS) directory. A
maximum of 1,000 items under the directory can be displayed. To display
the files and folders under the directory, add a slash (/) at the end of
these files and folders.

    :param bucket:          COS bucket.
    :type bucket:           str
    :param path:            path.
    :type path:             str
    :return:                directory information.
    :rtype:                 dict
    :param maker:           List entries starting from marker.
    :type maker:            str
    :param max_keys:        Configure the maximum number of entries
returned at a time. The maximum value is 1000.
    :type max_keys:         int
    :param encoding_type:   Configure the encoding method for returned
results. You can only set it to url.
    :type encoding_type:    str

upload_to_cos(self, local_path, bucket, cos_path)
    Upload files or directories from a local path to COS.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param local_path:  local path.
    :type local_path:   str
    :param bucket:      COS bucket.
    :type bucket:       str
    :param cos_path:    COS path.
    :type cos_path:     str
    :return:            None. Error messages are given through raise.
    :rtype:

download_from_cos(self, bucket, cos_path, local_path)
    Download files or directories from COS to a local machine.
    Note: If local files exist, they will be directly overwritten. When
cos_path is a directory and local_path is an existing directory, the
folder name of cos_path will be retained as a subdirectory.

    :param bucket:      COS bucket.
    :type bucket:       str
    :param cos_path:    COS path.
    :type cos_path:     str
    :param local_path:  local path.
    :type local_path:   str
    :return:            None. Error messages are given through raise.
    :rtype:


delete_cos_path(self, bucket, delete_path)
    Delete the COS directory. To delete files and folders under the
directory, add a slash (/) at the end of these files and folders. That is,
files and folders without a slash (/) will be deleted as files, and files
and folders with a slash (/) will be deleted as folders.

    :param bucket:      COS bucket.
    :type bucket:       str
    :param delete_path: path to be deleted.
    :type delete_path:  str
```

3. Methods Called Through HiveInitial

3.1. Initialization

©2013-2026 Tencent Cloud. All rights reserved.

```
from tikit.client import Client
from tikit.hive import HiveInitial

client = Client("your_secret_id", "your_secret_key", "<region>")

hive_init = HiveInitial(client)
hive_init.hive_initial("emr-xsjhbhf", "hadoop", "./emr.keytab")
```

3.2. APIs

```
  def spark_hive_initial_wedata(self, wedata_id, source_account=None):
      """Initialize WeData Hive for Spark. (After calling this method, you
can use Spark to perform Hive operations.)

      :param wedata_id:       WeData data source ID.
      :type wedata_id:        int
      :param source_account:     If Hive is a system source, the account
UIN needs to be specified.
      :type source_account:      str
      :rtype:
      """

  def hive_initial_wedata(self, wedata_id, source_account=None):
      """Obtain the WeData Hive connection handle.

      :param wedata_id:       WeData data source ID.
      :type wedata_id:        int
      :param source_account:     If Hive is a system source, the account
UIN needs to be specified.
      :type source_account:      str
      :rtype:
      """

  def spark_hive_initial(self, emr_id, username=None, keytab=None):
      """Initialize EMR Hive for Spark. (After calling this method, you
can use Spark to perform Hive operations.)

      :param emr_id:       Tencent Cloud EMR ID.
      :type emr_id:        str
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        :param username:     If Kerberos authentication is used, the
corresponding username needs to be specified.
        :type username:      str
        :param keytab:       keytab file path. If the default account (for
example, hadoop) of a cluster is used, the keytab path needs to be
provided.
        :type keytab:        str
        :rtype:
        """

    def hive_initial(self, emr_id, username=None, keytab=None):
        """Obtain the EMR Hive connection handle.

        :param emr_id:       Tencent Cloud EMR ID.
        :type emr_id:        str
        :param username:     If Kerberos authentication is used, the
corresponding username needs to be specified.
        :type username:      str
        :param keytab:       keytab file path. If the default account (for
example, hadoop) of a cluster is used, the keytab path needs to be
provided.
        :type keytab:        str
        :rtype:
        """

    def hive_initial_custom(
            self,
            host=None,
            port=None,
            scheme=None,
            username=None,
            database='default',
            auth=None,
            configuration=None,
            kerberos_service_name=None,
            password=None,
            check_hostname=None,
            ssl_cert=None,
            thrift_transport=None):
        """Connect to HiveServer2
```

©2013-2026 Tencent Cloud. All rights reserved.

```
      :param host: What host HiveServer2 runs on
      :param port: What port HiveServer2 runs on. Defaults to 10000.
      :param auth: The value of hive.server2.authentication used by
HiveServer2.
          Defaults to ``NONE``.
      :param configuration: A dictionary of Hive settings (functionally
same as the `set` command)
      :param kerberos_service_name: Use with auth='KERBEROS' only
      :param password: Use with auth='LDAP' or auth='CUSTOM' only
      :param thrift_transport: A ``TTransportBase`` for custom advanced
usage.
          Incompatible with host, port, auth, kerberos_service_name, and
password.

      The way to support LDAP and GSSAPI is originated from
cloudera/Impyla:

https://github.com/cloudera/impyla/blob/255b07ed973d47a3395214ed92d35ec061
5ebf62
      /impala/_thrift_api.py#L152-L160
      """

  def upload_to_wedata_hive(self, wedata_id, local_path, table_name,
database='default', overwrite=False,
                            partition='', source_account=None):
      """Upload files to WeData Hive.

      :param wedata_id:       WeData data source ID.
      :type wedata_id:        int
      :param local_path:     local file path.
      :type local_path:      str
      :param table_name:     table name.
      :type local_path:      str
      :param database:     database.
      :type database:      str
      :param overwrite:       Whether to delete the original data.
      :type overwrite:        bool
      :param partition:       partition selection.
      :type partition:        str
      :param source_account:     If Hive is a system source, the account
UIN needs to be specified.
```

```
        :type source_account:       str
        :rtype:
        """

    def export_from_wedata_hive(self, wedata_id, local_path, table_name='',
database='default', sql='',
                                row_format="row format delimited fields
terminated by ','", source_account=None):
        """Export WeData Hive data to a local machine.

        :param wedata_id:       WeData data source ID.
        :type wedata_id:        int
        :param local_path:      local file path.
        :type local_path:       str
        :param table_name:      table name.
        :type local_path:       str
        :param database:      database.
        :type database:       str
        :param sql:             SQL statement for querying data, for
example, select * from t1.
        :type sql:              str
        :param row_format:      row output format.
        :type row_format:       str
        :param source_account:      If Hive is a system source, the account
UIN needs to be specified.
        :type source_account:       str
        :rtype:
        """
```

©2013-2026 Tencent Cloud. All rights reserved.

Submitting a Training Task

Last updated: 2026-02-26 15:37:15

```
describe_postpaid_training_price(self)
    Query the hourly price for each configuration. Unit: CNY.

    :rtype:
tencentcloud.tione.v20211111.models.DescribeBillingSpecsResponse

describe_system_reasoning_images(self)
    Obtain a built-in inference image of the platform.

    :return:    inference image information.
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeInferTemplatesRe
sponse`
    The data format is as follows:
    {
        "FrameworkTemplates": [
        {
            "Framework": "TENSORFLOW",
            "FrameworkVersion": "2.4",
            "Groups": [
            "TENSORFLOW",
            "LIGHT"
            ],
            "InferTemplates": [
            {
                "InferTemplateId": "tf2.4-py38-cpu",
                "InferTemplateImage": "ccr.ccs.tencentyun.com/qcloud-ti-
platform/ti-cloud-infer-tensorflow-cpu:py38-tensorflow2.4-cpu-20211206"
            },
            {
                "InferTemplateId": "tf2.4-py38-gpu",
                "InferTemplateImage": "ccr.ccs.tencentyun.com/qcloud-ti-
platform/ti-cloud-infer-tensorflow-gpu:py38-tensorflow2.4-cu110-20211206"
            }
            ]
        }
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        ],
        "RequestId": "3654e19b-c2ba-4953-b131-d66495723008"
    }


def create_training_task(self,
                         name,
                         framework,
                         cos_output,
                         worker_resource,
                         code_package_path,
                         ps_resource=None,
                         input_data_config=None,
                         worker_start_cmd=None,
                         ps_start_cmd=None,
                         tags=None,
                         tuning_parameters_dict={},
                         resource_group_id="",
                         remark=None,
                         log_enable=False,
                         log_logset_id=None,
                         log_topic_id=None,
                         vpc_id=None,
                         sub_net_id=None,
                         restart_limit=0):
    """Create a training task.

    :param name:         task name.
    :type name:          str
    :param framework:    framework environment that is running.
    :type framework:     :class:`tikit.models.FrameworkInfo`
    :param cos_output:           output Cloud Object Storage (COS)
information.
    :type cos_output:            str
    :param worker_resource:      worker node configuration.
    :type worker_resource:       :class:`tikit.models.ResourceConfigInfo`
    :param code_package_path:    COS information on the code.
    :type code_package_path:     str
    :param ps_resource:          PS node configuration.
    :type ps_resource:           :class:`tikit.models.ResourceConfigInfo`
    :param input_data_config:    input data information.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :type input_data_config:    list or
:class:`tikit.models.TrainingDataConfig`
    :param worker_start_cmd:    worker startup command.
    :type worker_start_cmd:     str
    :param ps_start_cmd:        PS node startup command.
    :type ps_start_cmd:         str
    :param tags:                tags.
    :type tags:                 list of
:class:`tikit.tencentcloud.tione.v20211111.models.Tag`
    :param tuning_parameters_dict:  tuning parameter dictionary.
    :type tuning_parameters_dict:   dict
    :param resource_group_id:   prepaid resource group ID.
    :type resource_group_id:    str
    :param remark:              description.
    :type remark:               str
    :param log_enable:          log switch.
    :type log_enable:           bool
    :param log_logset_id:       logset ID.
    :type log_logset_id:        str
    :param log_topic_id:        log topic ID.
    :type log_topic_id:         str
    :param vpc_id:              Virtual Private Cloud (VPC) ID.
    :type vpc_id:               str
    :param sub_net_id:          subnet ID.
    :type sub_net_id:           str
    :param restart_limit:       maximum number of restart attempts for the
current task, up to 10. If the number of restart attempts exceeds this
limit, the task will be marked as abnormal.
    :type restart_limit:          int
    :return:
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.CreateTrainingTaskRespon
se`
    """

def describe_training_frameworks(self):
    """View a training framework.

    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeTrainingFramewor
ksResponse`
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    """

def describe_training_tasks(self, filters=None, tag_filters=None,
offset=0, limit=50, order="DESC",
                            order_field="UpdateTime"):
    """Obtain the training task list.

    :param filters:     filters. Example: [{ "Name": "TaskStatus",
"Values": ["Running"] }].
    :type filters:      list of Filter
    :param tag_filters: tag filters. Example: [{ "TagKey": "TagKeyA",
"TagValue": ["TagValueA"] }].
    :type tag_filters:  list of TagFilter
    :param offset:      offset. The default value is 0.
    :type offset:       int
    :param limit:       number of returned results. The default value is
50.
    :type limit:        int
    :param order:       sorting order of the output list. Valid values:
ASC: ascending order; DESC: descending order.
    :type order:        str
    :param order_field: Sorting is based on this field. Valid values:
CreateTime and UpdateTime.
    :type order_field:  str
    :return:
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeTrainingTasksRes
ponse`
    """

def describe_training_task(self, task_id)
    Obtain the information on a single training task.

    :param task_id: training task ID.
    :type task_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DescribeTrainingTaskResponse`

def describe_training_task_pods(self, task_id)
    Obtain the Pod list of a training task.
```

```
    :param task_id: training task ID.
    :type task_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DescribeTrainingTaskPodsRespon
se`

def describe_train_logs(self, pod_name, start_time=None, end_time=None,
limit=None, order=None,
                        context=None, filters=None):
    """View the logs of a training task.

    :param pod_name: Specifies the Pod of which you want to query its
logs. Wildcards are supported. To view the logs of all Pods for a training
task, enter <task_id>-*. Example: train-51cd6bf7ec1000-*.
    :type pod_name: str
    :param start_time: start time for log query. It is a time string in
RFC3339 format, for example, 2021-12-16T13:20:24+08:00. The default value
is one hour before the current time.
    :type start_time: str
    :param end_time: end time for log query. It is a time string in
RFC3339 format, for example, 2021-12-16T13:20:24+08:00. The default value
is the current time.
    :type end_time: str
    :param limit: number of logs you can query. The default value is 100,
and the maximum value is 100.
    :type limit: int
    :param order: sorting method. Valid values: ASC and DESC. The default
value is DESC.
    :type order: str
    :param context: pagination cursor.
    :type context: str
    :param filters: filters.
    :type filters: list of
tikit.tencentcloud.tione.v20211111.models.Filter
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeLogsResponse`

        If the returned object is not null, the next() method can be used
to continuously obtain the next page of logs (if there are multiple
pages), as shown in the following:
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        now_time = datetime.datetime.now(datetime.timezone.utc)
        now_time_str = now_time.isoformat()
        result = client.describe_train_logs("train-51cd6bf7ec1000-
37c5p5nlr01s-launcher",
                                            "2021-12-
10T09:32:03.823509+00:00",
                                            now_time_str,
                                            limit=30)
        print(result)
        print(result.next())
        print(result.next())
        print(result.next())
    """

def push_training_metrics(self, timestamp, value_map, task_id=None,
epoch=None, total_steps=None, step=None):
    """Report custom training metrics (a single metric). A sub-account can
call this parameter 20 times per second. You need to control the reporting
frequency in your training code to avoid exceeding the limit and
triggering errors. Alternatively, you can also use
push_training_metrics_list.

    :param timestamp:   timestamp.
    :type timestamp:    int
    :param value_map:   metric mapping. It indicates the mapping between
metric names and metric values.
    :type value_map:    map: str -> float
    :param task_id:     task ID. If it is left empty, the value of the
TI_TASK_ID environment variable of the current task node environment is
used.
    :type task_id:      str
    :param epoch:       epoch value.
    :type epoch:        int
    :param total_steps: total number of steps.
    :type total_steps:  int
    :param step:        step number.
    :type step:         int
    :return:
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.PushTrainingMetricsRespo
nse`
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    client.push_training_metrics(int(time.time()), {"field1": 11,
"field2": 12}, "task-id-00001", 3, 1000, 66)
    """

def push_training_metrics_list(self, metric_list)
    Report custom training metrics (list).

    :param metric_list: MetricData array. If Task ID is left empty, the
value of the TI_TASK_ID environment variable of the current task node
environment is used.
    :type metric_list:  list of
:class:`tencentcloud.tione.v20211111.models.MetricData`
    :return:
    :rtype:
:class:`tencentcloud.tione.v20211111.models.PushTrainingMetricsResponse`

def describe_training_metrics(self, task_id)
    Query custom training metrics.

    :param task_id: task ID.
    :type task_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DescribeTrainingMetricsRespons
e`

def stop_training_task(self, task_id)
    Stop a training task.

    :param task_id: training task ID.
    :type task_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.StopTrainingTaskResponse`

def delete_training_task(self, task_id)
    Delete a training task.

    :param task_id: training task ID.
    :type task_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DeleteTrainingTaskResponse`
```

©2013-2026 Tencent Cloud. All rights reserved.

```

```

Intermediate configurations:

```
tikit.models.FrameworkInfo

def new_custom(training_mode, image_type, image_url, registry_region=None,
registry_id=None,
                   user_name=None, passwd=None):
      """Configuration of a custom training framework. You can view the
list through describe_training_frameworks().

      :param training_mode:   training mode, for example, DDP. It only
needs to be configured for training tasks.
      :type training_mode:    str
      :param image_type:      image type. CCR refers to Tencent Container
Registry Individual Edition, TCR refers to Tencent Container Registry
Enterprise Edition, and CUSTOM indicates a third-party custom image.
      :type image_type:       str
      :param image_url:       image address. This parameter is required.
      :type image_url:        str
      :param registry_region: region of the image repository of TCR.
      :type registry_region:  str
      :param registry_id:     image repository ID of TCR.
      :type registry_id:      str
      :param user_name:       username of a custom image repository.
      :type user_name:        str
      :param passwd:          password of a custom image repository.
      :type passwd:           str
      :return:
      :rtype:
      """

def new_system_framework(framework_name, framework_environment,
training_mode):
    """Built-in training frameworks

    :param framework_name:      framework name. You can view the list
through describe_training_frameworks().
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :type framework_name:       str
    :param framework_environment:   framework environment. You can view
the list through describe_training_frameworks().
    :type framework_environment:    str
    :param training_mode:       training mode. You can view the list
through describe_training_frameworks().
    :type training_mode:        str
    :return:
    :rtype:
    """

-------------------------------------------------------
tikit.models.ResourceConfigInfo

def new_postpaid(instance_type, instance_num):
    """Obtain resource configurations in postpaid mode.

    :param instance_type:   instance type. You can view the instance list
through describe_postpaid_training_price().
    :type instance_type:    str
    :param instance_num:    number of instances.
    :type instance_num:     int
    :return:
    :rtype:
    """
|
def new_prepaid(cpu=0, memory=0, gpu=0, gpu_type=None, instance_num=1):
    """Obtain resource configurations in prepaid mode. When GPU-based
proportional presetting is enabled for a resource group, you can set the
cpu and memory parameters to 0 or do not configure them because the values
of the two parameters will be allocated proportionally based on the number
of GPU cards.

    :param cpu:     number of CPUs. Unit: core.
    :type cpu:      float
    :param memory:  memory size. Unit: GB.
    :type memory:   float
    :param gpu_type: GPU type.
    :type gpu_type: str
    :param gpu:     number of GPUs.
    :type gpu:      float
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param instance_num:    number of instances.
    :type instance_num:     int
    :return:
    :rtype:
    """

----------------------------------------------------
tikit.models.TrainingDataConfig

def new_mount_cos(cos_str, target_path):
    """One piece of training data of the COS type.

    :param cos_str:      COS storage. Format: <bucket>/<cos path>/.
    :type cos_str: str
    :param  target_path:  target mount path.
    :type target_path: str
    :return:
    :rtype:
    """

def new_dataset_mount(dataset_id, target_path):
    """One piece of training data of the dataset type.

    :param dataset_id:  data set ID.
    :type dataset_id: str
    :param  target_path:  target mount path.
    :type target_path: str
    :return:
    :rtype:
    """

def new_mount_cfs(cfs_id, source_path, target_path):
    """Create a training data set configuration of the Cloud File Storage
(CFS) type.

    :param cfs_id:      CFS instance ID.
    :type cfs_id: str
    :param  source_path: CFS path.
    :type source_path: str
    :param  target_path: target mount path.
    :type target_path: str
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :return:
    :rtype:
    """

def new_mount_hdfs(hdfs_id, source_path, target_path):
    """Create a training data set configuration of the HDFS type.

    :param hdfs_id:      HDFS ID on EMR.
    :type hdfs_id: str
    :param  source_path: HDFS path.
    :type source_path: str
    :param  target_path: target mount path.
    :type target_path: str
    :return:
    :rtype:
    """

def new_mount_wedata_hdfs(wedata_id, source_path):
    """Create a training data set configuration of the WeData HDFS type.

    :param wedata_id:     WeData data source ID.
    :type wedata_id: int
    :param  source_path: HDFS path.
    :type source_path: str
    :return:
    :rtype:
    """

def new_dataset(id_target_dict):
    """ Deprecated !
    Create a training data set configuration of the dataset type.

    :param id_target_dict:  data set information. dataset id indicates the
target download path.
    :type id_target_dict:   dict
    :return:
    :rtype:
    """

def new_cos_data(cos_str_target_dict):
    """Deprecated !
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    Create a training data set configuration of the COS type.

    :param cos_str_target_dict:     data set information. <bucket>/<cos
path>/ indicates the target download path.
    :type cos_str_target_dict:      dict
    :return:
    :rtype:
    """

def new_mount_goosefs(goosefs_id, source_path, namespace, target_path):
    """Create a training data set configuration of the GooseFS type.

    :param goosefs_id: GooseFS instance ID.
    :type goosefs_id: str
    :param  target_path: target mount path.
    :type target_path: str
    :return:
    :rtype:
    """

def new_mount_goosefsx(goosefsx_id, goosefsx_path, target_path):
    """Create a storage configuration of the Data Accelerator Goose
FileSystem (GooseFSx) type.

    :param goosefsx_id: GooseFSx instance ID.
    :type goosefsx_id: str
    :param goosefsx_path: GooseFSx path.
    :type goosefsx_path: str
    :param  target_path: target mount path.
    :type target_path: str
    :return:
    :rtype:
    """
```

©2013-2026 Tencent Cloud. All rights reserved.

Creating and Managing Dev Machines

Last updated: 2026-02-26 15:38:05

> **Note:**
> The Notebook module has been upgraded to the Dev Machines module on February 25, 2025, and Tikit-related APIs currently remain unchanged.

```
def create_notebook(self,
                    name,
                    image_info=None,
                    resource_info=None,
                    input_data_config=None,
                    ssh_config=None,
                    tags=None,
                    resource_group_id="",
                    lifecycle_script_id="",
                    code_repo_id_list=None,
                    auto_stop_enable=False,
                    auto_stop_time=None,
                    log_enable=False,
                    log_logset_id=None,
                    log_topic_id=None,
                    vpc_id=None,
                    subnet_id=None):
        """Create a Notebook instance.
        :param name:        instance name.
        :type name:         str
        :param image_info:   instance image information.
        :type image_info:    :class:`tikit.models.ImageInfo`
        :param resource_info:     instance resource configuration.
        :type resource_info:      :class:`tikit.models.ResourceConfigInfo`
        :param input_data_config:   input storage information.
        :type input_data_config:    list or
:class:`tikit.models.NotebookDataConfig`
        :param ssh_config:  SSH configuration information.
        :type ssh_config:   :class:`tikit.models.SSHConfig`
        :param tags:                tags.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        :type tags:                 list of
:class:`tikit.tencentcloud.tione.v20211111.models.Tag`
        :param resource_group_id: prepaid resource group ID.
        :type resource_group_id:    str
        :param lifecycle_script_id: lifecycle script ID.
        :type lifecycle_script_id:  str
        :param code_repo_id_list:   Git Repositories ID list.
        :type code_repo_id_list:    list
        :param auto_stop_enable:    automatic stop switch.
        :type auto_stop_enable:     bool
        :param auto_stop_time:      automatic stop time. Unit: hour.
        :type auto_stop_time:       int
        :param log_enable:          log switch.
        :type log_enable:           bool
        :param log_logset_id:       logset ID.
        :type log_logset_id:        str
        :param log_topic_id:        log topic ID.
        :type log_topic_id:         str
        :param vpc_id:   vpc_id that needs to be connected when you mount
Cloud File Storage (CFS) or Turbo File Storage (TurboFS).
        :type vpc_id:    str
        :param subnet_id:   subnet in the Virtual Private Cloud (VPC)
network when you mount CFS or TurboFS.
        :type subnet_id:    str
        :return:
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.CreateTrainingTaskRespon
se`
        """

def modify_notebook(self,
                    notebook_id,
                    image_info=None,
                    resource_info=None,
                    input_data_config=None,
                    ssh_config=None,
                    resource_group_id=None,
                    lifecycle_script_id=None,
                    code_repo_id_list=None,
                    auto_stop_enable=False,
                    auto_stop_time=None,
```

©2013-2026 Tencent Cloud. All rights reserved.

```
                    log_enable=None,
                    log_logset_id=None,
                    log_topic_id=None,
                    vpc_id=None,
                    subnet_id=None):
    """Edit a Notebook instance.
    :param notebook_id:  instance ID.
    :type notebook_id:   str
    :param image_info:   instance image information.
    :type image_info:    :class:`tikit.models.ImageInfo`
    :param resource_info:     instance resource configuration.
    :type resource_info:      :class:`tikit.models.ResourceConfigInfo`
    :param input_data_config:   input storage information.
    :type input_data_config:    list or
:class:`tikit.models.NotebookDataConfig`
    :param ssh_config:  SSH configuration information.
    :type ssh_config:   :class:`tikit.models.SSHConfig`
    :param resource_group_id:   prepaid resource group ID.
    :type resource_group_id:    str
    :param lifecycle_script_id: lifecycle script ID.
    :type lifecycle_script_id:  str
    :param code_repo_id_list:   Git Repositories ID list.
    :type code_repo_id_list:    list
    :param auto_stop_enable:    automatic stop switch.
    :type auto_stop_enable:     bool
    :param auto_stop_time:      automatic stop time. Unit: hour.
    :type auto_stop_time:       int
    :param log_enable:          log switch.
    :type log_enable:           bool
    :param log_logset_id:       logset ID.
    :type log_logset_id:        str
    :param log_topic_id:        log topic ID.
    :type log_topic_id:         str
    :param vpc_id:   vpc_id that needs to be connected when you mount CFS
or TurboFS.
    :type vpc_id:    str
    :param subnet_id:   subnet in the VPC network when you mount CFS or
TurboFS.
    :type subnet_id:    str
    :return:
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.CreateTrainingTaskRespon
se`
    """

def stop_notebook(self, notebook_id):
    """Stop a Notebook instance.

    :param notebook_id: notebookID
    :type notebook_id: str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.StopNotebookResponse`
    """

def delete_notebook(self, notebook_id):
    """Delete a Notebook instance.

    :param notebook_id: notebookID
    :type notebook_id: str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DeleteNotebookResponse`
    """

def start_notebook(self, notebook_id):
    """Start a Notebook instance.

    :param notebook_id: notebookID
    :type notebook_id: str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.StartNotebookResponse`
    """

def describe_notebooks(self, filters=None, tag_filters=None, offset=0,
limit=10, order="DESC",
                           order_field="UpdateTime"):
    """Obtain the Notebook list.

    :param filters:     filters. Example: [{ "Name": "Status", "Values":
["Running"], "Fuzzy": False}].
    Name (Notebook name): notebook1. Fuzzy matching is supported.
    Id (Notebook ID): nb-123456789. Fuzzy matching is supported.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    Status:
Submitting/Starting/Running/Stopped/Stopping/Failed/SubmitFailed/ImageSavi
ng.
    ChargeType (billing type): PREPAID (prepaid)/POSTPAID_BY_HOUR
(postpaid).
    ChargeStatus (billing status): NOT_BILLING (billing not
started)/BILLING (billing)/BILLING_STORAGE (storage billing)/ARREARS_STOP
(stopped due to overdue payments).
    :type filters:      list of Filter
    :param tag_filters: tag filters. Example: [{ "TagKey": "TagKeyA",
"TagValue": ["TagValueA"] }].
    :type tag_filters:  list of TagFilter
    :param offset:      offset. The default value is 0.
    :type offset:       int
    :param limit:       number of returned results. The default value is
10.
    :type limit:        int
    :param order:       sorting order of the output list. Valid values:
ASC: ascending order; DESC: descending order.
    :type order:        str
    :param order_field: Sorting is based on this field. Valid values:
CreateTime and UpdateTime.
    :type order_field:  str
    :return:
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeTrainingTasksRes
ponse`
    """

def describe_notebook(self, notebook_id):
    """Obtain the information on a single Notebook task.

    :param notebook_id: notebookID
    :type notebook_id: str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeNotebookkRespons
e`
    """

def describe_notebook_buildin_images(self):
    """Obtain the information on a single Notebook task.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param notebook_id: notebookID
    :type notebook_id: str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeBuildInImagesRes
ponse`
    """
```

©2013-2026 Tencent Cloud. All rights reserved.

Registering a Model

Last updated: 2026-03-04 18:58:28

```
create_model_by_cos(self, training_model_name, algorithm_framework,
model_cos_path, model_index, reasoning_env, delete_task_cos_model=True,
tags=None)
    Use the existing Cloud Object Storage (COS) path to create a model.

    :param training_model_name: model name.
    :type training_model_name: str
    :param algorithm_framework: algorithm framework.
    :type algorithm_framework: str
    :param model_cos_path: model COS directory, ending with a slash (/).
Format: <bucket>/<cos path>/.
    :type model_cos_path:  str
    :param model_index: training metric.
    :type model_index: str
    :param reasoning_env: running environment for inference.
    :type reasoning_env:  :class:`tikit.models.ReasoningEnvironment`
    :param delete_task_cos_model: Whether to delete the originally output
model files of a task.
    :type delete_task_cos_model: bool
    :param tags: tag configuration.
    :type tags: list of :class:`tencentcloud.tione.v20211111.models.Tag`
    :return:
    :rtype:
:class:`tencentcloud.tione.v20211111.models.CreateTrainingModelResponse`
    The returned content is as follows:
    {
      "Id": "m-23054253294030848",
      "RequestId": "c8f9b70e-bf4d-4c34-9480-21d902d3b341"
    }

create_model_by_task(self, training_model_name, training_job_id,
reasoning_env, training_model_index=None, delete_task_cos_model=True,
tags=None)
    Use the existing task results to create a model.

    :param training_model_name: model name.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :type training_model_name: str
    :param training_job_id: training task ID.
    :type training_job_id: str
    :param reasoning_env: running environment for inference.
    :type reasoning_env:  :class:`tikit.models.ReasoningEnvironment`
    :param training_model_index: training metric. If it is specified, the
value in the training task will be overwritten.
    :type training_model_index: str
    :param delete_task_cos_model: Whether to delete the originally output
model files of a task.
    :type delete_task_cos_model: bool
    :param tags: tag configuration.
    :type tags: list of :class:`tencentcloud.tione.v20211111.models.Tag`
    :return:
    :rtype:
:class:`tencentcloud.tione.v20211111.models.CreateTrainingModelResponse`
    The returned content is as follows:
    {
      "Id": "m-23054253294030848",
      "RequestId": "c8f9b70e-bf4d-4c34-9480-21d902d3b341"
    }

create_model_version_by_cos(self, training_model_id, algorithm_framework,
model_cos_path, model_index, reasoning_env, delete_task_cos_model=True,
tags=None)
    Use the existing COS path to create a model version.

    :param training_model_id: model ID.
    :type training_model_id: str
    :param algorithm_framework: algorithm framework.
    :type algorithm_framework: str
    :param model_cos_path: model COS directory, ending with a slash (/).
Format: <bucket>/<cos path>/.
    :type model_cos_path:  str
    :param model_index: training metric.
    :type model_index: str
    :param reasoning_env: running environment for inference.
    :type reasoning_env:  :class:`tikit.models.ReasoningEnvironment`
    :param delete_task_cos_model: Whether to delete the originally output
model files of a task.
    :type delete_task_cos_model: bool
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param tags: tag configuration.
    :type tags: list of :class:`tencentcloud.tione.v20211111.models.Tag`
    :return:
    :rtype:
:class:`tencentcloud.tione.v20211111.models.CreateTrainingModelResponse`
    The returned content is as follows:
    {
      "Id": "m-23054253294030848",
      "RequestId": "c8f9b70e-bf4d-4c34-9480-21d902d3b341"
    }

create_model_version_by_task(self, training_model_id, training_job_id,
reasoning_env, training_model_index=None, delete_task_cos_model=True,
tags=None)
    Use the existing COS path to create a model version.

    :param training_model_id: model ID.
    :type training_model_id: str
    :param training_job_id: training task ID.
    :type training_job_id: str
    :param reasoning_env: running environment for inference.
    :type reasoning_env:  :class:`tikit.models.ReasoningEnvironment`
    :param training_model_index: training metric. If it is specified, the
value in the training task will be overwritten.
    :type training_model_index: str
    :param delete_task_cos_model: Whether to delete the originally output
model files of a task.
    :type delete_task_cos_model: bool
    :param tags: tag configuration.
    :type tags: list of :class:`tencentcloud.tione.v20211111.models.Tag`
    :return:
    :rtype:
:class:`tencentcloud.tione.v20211111.models.CreateTrainingModelResponse`
    The returned content is as follows:
    {
      "Id": "m-23054253294030848",
      "RequestId": "c8f9b70e-bf4d-4c34-9480-21d902d3b341"
    }

describe_training_model_version(self, training_model_version_id)
    View the information on a single version.
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param training_model_version_id: model version ID.
    :type training_model_version_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DescribeTrainingModelVersionRe
sponse`

describe_training_model_versions(self, training_model_id)
    View the information list of each version of a model.

    :param training_model_id: model ID.
    :type training_model_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DescribeTrainingModelVersionsR
esponse`

describe_training_models(self, filters=None, order_field=None, order=None,
offset=None, limit=None, tag_filters=None)
    View the model list.

    :param filters: filters.
    :type filters: list of Filter
    :param order_field: sorting field.
    :type order_field: str
    :param order: sorting method. Valid values: ASC and DESC.
    :type order: str
    :param offset: offset.
    :type offset: int
    :param limit: number of returned results.
    :type limit: int
    :param tag_filters: tag filters.
    :type tag_filters: list of TagFilter
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DescribeTrainingModelsResponse
`
    print: The returned results are as follows:
    +--------------------+----------------------+-------------------+---
-------------------+
    |        Model ID        |          Name          |        Tag        |
Creation Time       |
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    +--------------------+----------------------+-------------------+---
-------------------+
    | m-23054253294030848 | tikit-model-task-1227 |                    |
2021-12-27T14:22:43Z |
    | m-23054252760240128 |  tikit-model-cos-1227 |                    |
2021-12-27T14:22:35Z |
    | m-23054246746066944 |    tikit-model-task   |                    |
2021-12-27T14:21:03Z |
    | m-23037023501881344 |    tikit-model-cos    |                    |
2021-12-24T13:20:57Z |
    | m-23036973226987520 |   tikit-model-name-2  |                    |
2021-12-24T13:08:10Z |
    | m-23034443699064832 |      model_cos-2      |                    |
2021-12-24T02:24:52Z |
    | m-23028904650346496 |       model_cos       | tag_001:tag_v_001 |
2021-12-23T02:56:13Z |
    | m-23028894739075072 |       model_task      |                    |
2021-12-23T02:53:42Z |
    |  22997374387884032  |         xx004         |                    |
2021-12-17T13:17:39Z |
    |  22996889833377792  |         xx003         |                    |
2021-12-17T11:14:26Z |
    +--------------------+----------------------+-------------------+---
-------------------+

delete_training_model(self, training_model_id)
    Delete a model.

    :param training_model_id: model ID.
    :type training_model_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DeleteTrainingModelResponse`

delete_training_model_version(self, training_model_version_id)
    Delete a model version.

    :param training_model_version_id: model version ID.
    :type training_model_version_id: str
    :rtype:
:class:`tencentcloud.tione.v20211111.models.DeleteTrainingModelVersionResp
onse`
```

©2013-2026 Tencent Cloud. All rights reserved.



©2013-2026 Tencent Cloud. All rights reserved.

Model Optimization

Last updated: 2026-03-04 18:58:59

```
def create_model_accelerate_task(self,
                         model_acc_task_name,
                         model_source,
                         model_format,
                         model_name,
                         model_version,
                         model_input_path,
                         model_output_path,
                         tensor_infos,
                         acc_engine_version = None,
                         optimization_level = None,
                         gpu_type = None,
                         hyper_parameter = None,
                         tags = None,
                         model_signature = None):
    """Create a model acceleration task.

    :param model_acc_task_name:         task name.
    :type model_acc_task_name:          str
    :param model_source:                model source.
    :type model_source:                 str
    :param model_format:                model format
(TORCH_SCRIPT/DETECTRON2/SAVED_MODEL/FROZEN_GRAPH/MMDETECTION/ONNX/HUGGING
_FACE).
    :type model_format:                 str
    :param model_name:                  model name.
    :type model_name:                   str
    :param model_version:               model version.
    :type model_version:                str
    :param model_input_path:            input Cloud Object Storage (COS)
path of a model.
    :type model_input_path:             :class:`tikit.models.CosPathInfo`
    :param model_output_path:           output COS path of an optimized
model.
    :type model_output_path:            :class:`tikit.models.CosPathInfo`
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param acc_engine_version:          optimization engine version. If
this parameter is not specified, the latest version is used by default.
    :type acc_engine_version:           str
    :param tensor_infos:                Tensor information.
    :type tensor_infos:                 list of str
    :param optimization_level:          optimization level (NO_LOSS/FP16).
FP16 is used by default.
    :type optimization_level:           str
    :param gpu_type:                    GPU card type (T4/V100). T4 is used
by default.
    :type gpu_type:                     str
    :param hyper_parameter:             hyperparameter settings.
    :type hyper_parameter:
:class:`tikit.tencentcloud.tione.v20211111.models.HyperParameter`
    :param tags:                        tags.
    :type tags:                         list of
:class:`tikit.tencentcloud.tione.v20211111.models.Tag`
    :param model_signature:             signature configured when
SAVED_MODEL is saved.
    :type model_signature:              str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.CreateModelAccelerateTas
kResponse`
    """

def stop_model_accelerate_task(self, model_acc_task_id):
    """Stop a model acceleration task.

    :param model_acc_task_id:         model acceleration task ID.
    :type model_acc_task_id:          str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.StopModelAccelerateTaskR
esponse`
    """

def delete_model_accelerate_task(self, model_acc_task_id):
    """Delete a model acceleration task.

    :param model_acc_task_id:         model acceleration task ID.
    :type model_acc_task_id:          str
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DeleteModelAccelerateTas
kResponse`
    """

def restart_model_accelerate_task(self,
                                  model_acc_task_id,
                                  model_source,
                                  model_format,
                                  model_name,
                                  model_version,
                                  model_input_path,
                                  model_output_path,
                                  tensor_infos,
                                  acc_engine_version = None,
                                  optimization_level = None,
                                  gpu_type = None,
                                  hyper_parameter = None,
                                  tags = None,
                                  model_signature = None):
    """Restart a model acceleration task.

    :param model_acc_task_id:        task ID.
    :type model_acc_task_id:         str
    :param model_source:             model source.
    :type model_source:              str
    :param model_format:             model format
(TORCH_SCRIPT/DETECTRON2/SAVED_MODEL/FROZEN_GRAPH/MMDETECTION/ONNX/HUGGING
_FACE).
    :type model_format:              str
    :param model_name:               model name.
    :type model_name:                str
    :param model_version:            model version.
    :type model_version:             str
    :param model_input_path:         input COS path of a model.
    :type model_input_path:          :class:`tikit.models.CosPathInfo`
    :param model_output_path:        output COS path of an optimized
model.
    :type model_output_path:         :class:`tikit.models.CosPathInfo`
    :param tensor_infos:             Tensor information.
    :type tensor_infos:              list of str
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param acc_engine_version:       optimization engine version. If this
parameter is not specified, the latest version is used by default.
    :type acc_engine_version:        str
    :param optimization_level:       optimization level (NO_LOSS/FP16).
FP16 is used by default.
    :type optimization_level:        str
    :param gpu_type:                 GPU card type (T4/V100). T4 is used
by default.
    :type gpu_type:                  str
    :param hyper_parameter:          hyperparameter settings.
    :type hyper_parameter:
:class:`tikit.tencentcloud.tione.v20211111.models.HyperParameter`
    :param tags:                     tags.
    :type tags:                      list of
:class:`tikit.tencentcloud.tione.v20211111.models.Tag`
    :param model_signature:          signature configured when SAVED_MODEL
is saved.
    :type model_signature:           str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.RestartModelAccelerateTa
skResponse`
    """

def describe_model_accelerate_tasks(self, filters=None, order_field=None,
order=None, offset=None, limit=None,
                                    tag_filters=None):
    """View the list of acceleration tasks.

    :param filters:         filters.
    :type filters:          list of filters. ModelAccTaskName/ModelSource
is supported.
    :param order_field:     sorting field. CreateTime is used by default.
    :type order_field:      str
    :param order:           sorting method. Valid values: ASC and DESC.
DESC is used by default.
    :type order:            str
    :param offset:          offset.
    :type offset:           int
    :param limit:           number of returned results. The default value
is 10.
    :type limit:            int
```

©2013-2026 Tencent Cloud. All rights reserved.

```
    :param tag_filters:     tag filters.
    :type tag_filters:      list of TagFilter
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeModelAccelerateT
asksResponse`
    """

def describe_model_accelerate_task(self, model_acc_task_id):
    """Query the details of a model acceleration task.

    :param model_acc_task_id:        model acceleration task ID.
    :type model_acc_task_id:         str
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeModelAccelerateT
askResponse`
    """

def create_optimized_model(self, model_acc_task_id, tags = None):
    """Save an optimized model.

    :param model_acc_task_id:        model acceleration task ID.
    :type model_acc_task_id:         str
    :param tags:                     tags.
    :type tags:                      list of
:class:`tikit.tencentcloud.tione.v20211111.models.Tag`
    :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.CreateOptimizedModelResp
onse`
    """
```

©2013-2026 Tencent Cloud. All rights reserved.

Submitting a Batch Task

Last updated: 2026-03-10 10:16:29

```
create_batch_task(self,
                          batch_task_name,
                          job_type,
                          resource_config_info,
                          output_config,
                          input_config,
                          resource_group_id=None,
                          model_info=None,
                          image_info=None,
                          code_package=None,
                          start_cmd=None,
                          cron_info=None,
                          tags=None,
                          log_enable=False,
                          log_logset_id=None,
                          log_topic_id=None,
                          vpc_id=None,
                          subnet_id=None,
                          remark=""):
        """Create a batch task.

        :param batch_task_name: name of a batch task.
        :type batch_task_name: str
        :param job_type: job type. 1: one-time; 2: periodic.
        :param job_type: int
        :param resource_config_info: resource configuration.
        :type resource_config_info:
:class:`tikit.models.ResourceConfigInfo`
        :param output_config: output data source configuration. (If it is
set to COS, the target path must be an empty string.)
        :type output_config: list or
:class:`tikit.models.TrainingDataConfig`
        :param input_config: input data source configuration. (If it is
set to COS, the target path must be an empty string.)
        :type input_config: list or
:class:`tikit.models.TrainingDataConfig`
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        :param resource_group_id: prepaid resource group ID.
        :type resource_group_id:    str
        :param model_info: model information.
        :type model_info: :class:`tikit.models.ModelInfo`
        :param image_info: image information. (If a model file is used,
the custom image will be invalid.)
        :type image_info: :class:`tikit.models.FrameworkInfo`
        :param code_package: Cloud Object Storage (COS) information on the
code.
        :type code_package: str
        :param start_cmd: startup command.
        :type start_cmd: str
        :param cron_info: cycle configuration information.
        :type cron_info:
:class:`tikit.tencentcloud.tione.v20211111.models.CronInfo`
        :param tags: tags.
        :type tags: list of
:class:`tikit.tencentcloud.tione.v20211111.models.Tag`
        :param log_enable: log switch.
        :type log_enable: bool
        :param log_logset_id: logset ID.
        :type log_logset_id: str
        :param log_topic_id: log topic ID.
        :type log_topic_id: str
        :param vpc_id: vpc id
        :type vpc_id: str
        :param subnet_id:
        :type subnet_id: subnet ID.
        :param remark: task description.
        :type remark: str
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.CreateBatchTaskResponse`
        """

describe_batch_task(self, batch_task_id):
        """Query batch tasks.

        :param batch_task_id:
        :type batch_task_id: str
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeBatchTaskRespons
```

©2013-2026 Tencent Cloud. All rights reserved.

```
e`
        """

describe_batch_tasks(self, filters=None, tag_filters=None, offset=0,
limit=50, order="DESC",
                             order_field="UpdateTime"):
        """Obtain the list of batch tasks.

        :param filters:     filters. Example: [{ "Name": "TaskStatus",
"Values": ["Running"] }].
        :type filters:      list of Filter
        :param tag_filters: tag filters. Example: [{ "TagKey": "TagKeyA",
"TagValue": ["TagValueA"] }].
        :type tag_filters:  list of TagFilter
        :param offset:      offset. The default value is 0.
        :type offset:       int
        :param limit:       number of returned results. The default value
is 50.
        :type limit:        int
        :param order:       sorting order of the output list. Valid
values: ASC: ascending order; DESC: descending order.
        :type order:        str
        :param order_field: Sorting is based on this field. Valid values:
CreateTime and UpdateTime.
        :type order_field:  str
        :return:
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DescribeBatchTasksRespon
se`
        """

stop_batch_task(self, batch_task_id):
        """Stop a batch task.

        :param batch_task_id:
        :type batch_task_id: str
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.StopBatchTaskResponse`
        """

delete_batch_task(self, batch_task_id):
```

©2013-2026 Tencent Cloud. All rights reserved.

```
        """Delete a batch task.

        :param batch_task_id:
        :type batch_task_id: str
        :rtype:
:class:`tikit.tencentcloud.tione.v20211111.models.DeleteBatchTaskResponse`
        """
```

©2013-2026 Tencent Cloud. All rights reserved.