

Tencent Cloud TI Platform Getting Started Product Documentation





Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice

STencent Cloud

All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.



Contents

Getting Started

Signing Up and Applying for Permission

Data Center Overview

Model Building Through Task-based Modeling

Publishing and Calling Model Service

Getting Started with TiKit

Getting Started Signing Up and Applying for Permission

Last updated : 2024-07-22 17:44:13

To use the Tencent Cloud TI Platform, perform the following steps:

Step 1. Sign up for a Tencent Cloud account

Note:

If you already have a Tencent Cloud account, skip this step.

Step 2. Apply for access to Tencent Cloud TI Platform

Tencent Cloud TI Platform is in beta test and is open for application. Submit an application as follows: Click **Submit a Ticket** to go to the ticket system.

1. Select the ticket type.



Ticket				
Select your products	2 Select the issue type	>	3 Submit Ticket	
Cloud Virtual Machine Stable and secure and high) Iy easy-to-use computing s	⊡a	Tencent Real Time Communica Tencent Real-Time Communication, TRTC	I
▲ All categories of problems report	ted by tickets First, clic	k here fo	or more categories selection	
General problems	Computing and Network		Storage & CDN	Database
ICP license	Cloud Virtual Machine		Cloud Object Storage	CDB for MySQL
Account Center	Cloud Block Storage		Content Delivery Network	CDB for SQL server
Marketplace	Virtual Private Cloud		Cloud Image (CI)	PostgreSQL
	Cloud Load Balance		File storage CFS	Cloud Redis Storage
	Tencent Kubernetes Engine		Log Services CLS	Cloud MongoDB Servi
	Tencent Container Registry			Cloud Memcached (Cl
	Serverless Cloud Function			
Video and communication service	Other service			
Tencent Real Time Communication	Others Second, clic	k "Other	s"	
SMS				
VOD				
CSS				
Instant Messaging				
MPS				

2. Select the issue type.

Submit Ticket		
	Select your products > 2 Select the issue type	> (3) Submit Ti
	Problem Types	
	Functional consultation	Troubleshooting
	Third, click "Funcational consultation"	

3. Set information as required.



4. Submit the application. You can check the reply on this page.

Ticket Details	
	Current Status Not processed Ticket No.
	Communication records
	Problem Description TI test Confidential info: *****
	Tencent Cloud Engineer Tips: If you included confidential information such as your SecretId/SecretKey or account password when submitting a ticket, please modify these information once to
	Problem description(Required)
	Supported attachment format: .png, .jpg, .jpeg, .txt, .rar, .doc, .xls, .xlsx, .zip, .7z, .mp4; up to 50 MB
	Add attachments
	Enter confidential information (instance name, account, password) that requires encryption(optional)
	Submit

If you confirm from the reply that you have been granted access, you can access the console and search for Tencent Cloud TI Platform to go to its console.

Data Center Overview

Last updated : 2022-03-10 16:47:13

Overview

Data Center provides the following features:

Datasets

- All datasets under the current root account can be centrally managed and are isolated by region.
- Supported data types: Text (txt and csv), image (jpg, png, jpeg, and bmp), table (xls, xlsx and csv), and other (any format).
- You can preview data of the text, image, or table type.
- You can import image data labeled in the TI platform format or the industry common format PASCAL VOC or COCO. The imported labeling information can be visualized.

Data Labeling

- Provides a built-in scenario **Image classification**: Identifies the category (such as cat or dog) to which the image belongs.
- Provides a built-in scenario **Object detection**: Detects a specified object and returns label information, for example, detects a cat or dog and returns a box around each object.
- Provides a built-in scenario Object tracking: Tracks an object in multiple consecutive images.
- Provides a built-in scenario Image segmentation: Segments objects of a specified category.
- Provides a built-in scenario **OCR recognition**: Identifies text information.

Model Building Through Task-based Modeling

Last updated : 2022-02-25 17:46:30

Overview of Task-based Modeling

Task-based modeling provides model building capabilities in a wizard-like training task submission method and supports task submission based on multiple algorithm sources. It allows you to directly bind mainstream high-performance, distributed training frameworks through code package to quickly submit and start training tasks. The following uses a simple PyTorch MPI job as an example to demonstrate how to quickly create a task in task-based modeling.

Data Preparations

Dataset

This document uses the MNIST database available here.

Code package

The training script in this document is written by using the PyTorch framework. Its code package can be downloaded here.

Directions

Step 1 of task creation

1. Go to Training > Task-based Modeling and click New Task to enter the task creation wizard.

TI Platform		Task-based Modeling	Singapore	*										
🖬 Data Center 🔹 👻		 A training task in running 	status will incur c	costs. Stop a tas	k that is not need	ded.								
Training ^	۱	New Task								All	.	Enter a name		Q
Modeling - Notebook		Name	Training	Training	Billing M	Occupied Resource (j)		La ▼	Status T	Running	Traini 🗘	Monit Operat	ion	
🗳 Models 🔹 👻							No data yet							
Model Services *		Total items: 0									10 🔻 / pa	ge H 4 1	/1 page	► H



- 1. Enter the following information on the basic information page:
 - Task Name: mnist_train
 - Training Framework: select Built-in framework / PyTorch / 1.9-py3.6-cuda11.1-gpu.
 - Training Mode: select MPI.
 - Billing Mode: select Pay-as-You-Go.
 - Spec: select 8C40G V100*1.
 - Node Count: select 1.
 - Tag and Description: optional

After entering the required information, click **Next** to go to the task configuration page.

TI Platform	Create lask	
📩 Data Center 🛛 👻		
☐ Training ^	 A training task in ru 	inning status will incur costs. Stop a task that is not needed.
 Task-based Modeling 	1 Basic Informat	ion > (2) Task Configuration
 Notebook 	Task Name •	mnist_train 🥝
🗳 Models 🛛 👻		Max 60 characters; supports letters, digits, underscores (_), and hyphens (-); must start with a letter, digit, or Chinese character
Model Services ~	Region *	Singapore
	Training Framework *	Built-in / TensorFlow / 1.15-py3.6-cpu 💌
	Training Mode •	PS-Worker MPI Horovod
	Billing Mode •	Postpaid
	Spec •	8C40G V100*1 -
	Node Count *	- 1 +
	Label (j)	+ Add
	Description	Max 500 characters
	Configuration Price	
	Training task configuration	n price4.76 USD/hour
Ξ	Next Cancel	

Step 2 of task creation

Enter the following information on the task configuration page:

- 1. Algorithm Source:
 - Code Package: click **Select**. In the COS pop-up window, select the target bucket, click **Upload Folder** in the bottom-left corner, upload the mnist.pytorch folder generated after decompressing the code package to the bucket, and select the path of the code package.
 - Startup Commands: enter sh start.sh .



A training task in ru	Please select folder			\times
	Bucket List / demo-1309211636 🖋		Search files	Q,
Basic Informat	Object Name	Size	Last Updated	
orithm Source *	O D mnist.pytorch 2/	-	-	
	□ 🖿 ti-images 2/	-	-	
a Source 🚯				
lect Dataset 🛈				
na Daramatar 🔿				
ng Parameter 🕕				
	Upload Files			
	Upload Folder			
nfiguration Price	Create Folder			
aining task configuration	Upload Files *		OK Can	cel

2. Data Source: select **COS**.

- Mapping Path: enter train .
- Dataset Path: click Select. In the COS pop-up window, select the target bucket, click Upload Folder in the bottom-left corner, upload the ti-images folder generated after decompressing the dataset, and select the folder path as shown below:



Please select folder	
Please select folder	
	\times
Bucket List / demo-1309211636 💉 Search files	Q,
Basic Informat Object Name Size Last Updated	
m Source -	
O 🖬 ti-images 2/	
cos 🕕	
farameter 🕐	
Upload Files V Canc	el

TI Platform	← Create Task	
 Data Center ~ Training ^ 	 A training task in rur 	nning status will incur costs. Stop a task that is not needed.
Task-based Modeling	Basic Informati	ion > 2 Task Configuration
 Notebook 	Algorithm Source •	Code demo. 1309211838/minist nutrumh Select
🗳 Models 🗸 🗸		Package ① · Select a file in COS
Model Services *		Startup Enter a startup command Commands
	Data Source (j)	Dataset COS
	Select COS 🤅	Mapping Path () Dataset Path Operation
		/opt/ml/input/data/ train demo-1309211636/ti-images 2/ Select Delete
		+ Add
	Tuning Parameter (j)	1 Ω
	Configuration Price	
	Training task configuration	price4.76 USD/hour
Ξ	OK Previous	

- 3. Tuning Parameter: none.
- 4. Training Output: click **Select**. In the COS pop-up window, select the target bucket path for saving the training output data as shown below:



- 1. CLS Log: toggle it off.
- 2. VPC: select **Do not use VPC**.

After completing the configuration, view the hourly rate of the training task at the bottom of the page and click **OK**.

Publishing and Calling Model Service

Last updated : 2022-03-01 09:42:32

Publishing and Calling Model Service

Demo 1. Deploy a model trained on the platform as an online service

This demo uses the handwritten digit recognition model trained with the MNIST dataset and PyTorch framework in the task-based modeling module as an example to describe how to deploy a model trained on the platform as an online service and perform call verification.

Preparations

1. Model file

Log in to the TI Platform console, select **Training** > **Task-based Modeling** on the left sidebar, and confirm that the handwritten digit recognition model has been completely trained.

Task-based Modeling	Singapo	ore 🔻									
(i) A training task in running	status will in	cur costs.	Stop a task	that is not needed.							
New Task						All			Enter a	name	Q
Name	Trai	Trai	Billi	Occupied Resource (i)	Tag T	Status T	Run	Tr \$	Monitoring	Operation	
mnist_train	PyTorc h: 1.9- py3.6- cuda1 1.1- gpu	MPI	Postpa id	Spec: 8C40G V100*1 Node Count:1	Ø	Finished	3 mi	202	M D	Publish to Model Repositor Start More ▼	,

2. Inference script and configuration file

The inference script and configuration file of the handwritten digit recognition model have been prepared, which can be downloaded here.

3. Test data

The following test data has been prepared and can be directly used for call testing after the service is published.

- (1) Image of 2: https://qcloudimg.tencent-cloud.cn/raw/81ebbd130357ec4b1bbbecf73d1330f5.jpeg
- (2) Image of 5: https://qcloudimg.tencent-cloud.cn/raw/43b8f14ba3515c34b5c572f0c6bc225e.jpeg
- (3) Image of 9: https://qcloudimg.tencent-cloud.cn/raw/3768c5d8a47b5da666c82a286e00759b.jpeg



Step 1. Import the model repository

(1) Go to the TI Platform console and select **Models** > **Model Repository** on the left sidebar to enter the model list page.

(2) Click Import Model and set the following parameters in the Import Model pop-up:

- Import: select New model
- Model Name: customize the model name, such as mnist_train
- Tag: add optional tags as needed
- Model Source: select Task
- Task: select mnist_train
- Model Metrics: enter a metric as needed, such as accuracy
- Running Environment Source: we recommend you use a built-in GPU running environment, such as py1.9.0py36-cu111
- Move Model By: select a method as needed, such as Copy

Model Reposi	tory Singapore -		Down
Import Model	Import Model	>	Enter a model na
Model Nam	Import	New model New version	Operation
▶ mnist	Model Name *	mnist_train	Delete Edit Label
) pmml		Max 60 characters; supports letters, digits, underscores (_), and hyphens (-); must start with a letter or digit	Delete, Edit Label
	Tag 🛈	+ Add	
► Alina-test-0	Model Source	O Task ○ COS	Delete Edit Label
▶ test	Task *	mnist_train v	Delete Edit Label
Total items: 4	Algorithm Framework	PyTorch v	/ page
	Model Metrics	accuracy	
	Running Environment Source *	Built-in / py1.9.0-py36-cu111 🔹	
	Model File	sophietest-1309585087/output/ Select	
	Move Model By	Cut OCopy	
		OK Cancel	

(3) Click **OK**.

(4) Upload the inference script and configuration file.



Find the uploaded model version and click **Upload File**. In the COS pop-up window, click **Upload File** in the bottomleft corner, upload the 3 files generated after decompressing the <code>mnist-pytorch-infer</code> package successively, and click **OK**.

Model Repository	Singapore 🔻							Download Inference Code Ter	mplate
Import Model								Enter a model name	Q
Model Name		Tag T		Creation Tir	me 🕈			Operation	
▼ mnist_train		\bigcirc		2022-02-28	17:29:40			Delete Edit Label	
Model Version	Task Name	Algorithm Framework	Running Environment Source	Runnin	Model	COS P	Creatio	Operation	
v1	mnist_train 🗖	PyTorch	Built-in	ccr.c 🗗	accuracy	s. 🖬 🛃	2022-0	Upload File Publish Package Delete View Associated Service Group	



v-23409826560495616		Search files	Q
Object Name	Size	Last Updated	
≘ config.json	1.84KB	-	
mnist_cnn.pt	4.58MB	2022-02-28 17:29	
	5.7KB	-	
= net.py	776B	-	
iione_model_md5sums.txt	46B	2022-02-28 17:29	

Step 2. Start an online service

(1) Go to the TI Platform console and select **Model Services** > **Online Services** on the left sidebar to enter the service list page.

(2) Click **New Service** and enter the basic service information in the pop-up window as instructed below:

- Service Name: customize the service name, such as mnist-pytorch-infer
- Service Description: enter the optional service description as needed
- Billing Mode: if you have not used the resource group management module, we recommend you select Postpaid

Service Name *	mnist-pytorch-infer	\odot
	Max 60 characters; supports letters, digits, underscores (_), and hyphens (-); n	nust start with a letter or digit
Service Version	V1	
Service Description (optional)	Max 500 characters	
	0 / 500	
	~]
Region *	Singapore	
Billing Mode *	O Postpaid (pay-as-you-go)	

(3) Set the instance container information as instructed below:

- Use Model File: select Yes
- Model: select the model file <code>mnist_train</code> imported in the previous step
- Version: select v1
- Spec: select a spec as needed, such as 32C128G T4*1

Use Model File	No Yes	
Container Name	main	
Model *	mnist_train 🔹 🗘	
Version *	v1 •	
Running Environment *	ccr.ccs.tencentyun.com/qcloud-ti-platform/ti-cloud-infer-pytorch-gpu:py36	
Environment Version	py36-torch1.9.0-cu111-20211206	
Spec *	32C128G T4*1	
Environment Variable	+ Add	

(4) Complete the advanced settings of the service as instructed below:

- CLS Log Shipping: enable it as needed, which is disabled by default
- Instance Adjustment: select an option as needed, which is Manual by default
- Instance Count: configure a quantity as needed, which is 1 by default
- · Generate Authentication Token: enable it as needed, which is disabled by default
- Tag: add optional tags as needed

CLS Log Shipping 🛈	The TI console displays the logs of the last 7 days by default. If you want to store the logs persistently or use features such as log search, please enable CLS log shipping. For the CLS introduction and billing, view Document 🗳
Instance Adjustment	• Manual Set the number of instances • Auto Auto scales in if the system metric value is smaller than the configured value of the trigger metric, and auto scales out if it is larger than the set value
Instance Count *	- 1 +
Generate Authentication Token	Enable
Tag 🛈	+ Add

(5) Click Start Service. If you have selected the postpaid mode, a fee freezing confirmation will be displayed. The postpaid mode requires freezing the fees for two hours in advance. If your account balance is sufficient, click OK. In the service list, the newly created service will be in Creating status. A gateway will be created, and compute resources will be scheduled in the service deployment process, which will take some time. After the service is deployed successfully, its status will change to Running.

Step 3. Test the service call

(1) Confirm that the service published in the previous step has been deployed and is in **Running** status.(2) Click **Call** to enter the service call page.

Online Services	Singapore 🔻							
New Service							Enter a name	Q
Name	Status ▼	Billing Mode	Service ID	Running/Total Ver	Tag T	Creation Time	Coperation	
mnist-pytorch-infer	Running	Postpaid	ms-8zp8rltv	1/1	-	2022-02-28 17:41	1:39 Add Version Edit Label Call Del	lete

(3) View the API information and click **Online Testing** in the API list to open the service call testing page.



Туре	Operation
	Details Online Testing
1	ўуре

(4) Enter the request information in JSON format in the request body module, click **Send Request**, and you can view the prediction result in the response module.

• Sample 1. When the path {"images":["https://qcloudimg.tencent-

cloud.cn/raw/81ebbd130357ec4b1bbbecf73d1330f5.jpeg"]} of the image of 2 is entered in the request body, the prediction result will be 2.

Request	Body	Response
1	<pre>{"images":["https://qcloudimg. tencent-cloud.cn/raw/ 81ebbd130357ec4b1bbbecf73d1330f5.jpeg"] }</pre>	<pre>10 { 11 "result": { 12 "type": "classification", 13 "model_name": "mnist-classify", 14 "pred": 2 15 } 16 }</pre>

🔗 Tencent Cloud

• Sample 2. When the path {"images": ["https://qcloudimg.tencentcloud.cn/raw/43b8f14ba3515c34b5c572f0c6bc225e.jpeg"]} of the image of 5 is entered in the request body, the prediction result will be 5.

\$ ◎ ◎ □ ◎ □	E查看的是新版	反本 返回旧版			
← mi	nist-pytor	rch-in 在线测	则试	×	
服务管理	服务	调用请求体	k(Request Body)	请求响应(Response)	
调用监持		1	1 {"images":["https://qcloudimg. tencent-cloud.cn/raw/ 43b8f14ba3515c34b5c572f0c6bc225e.jpeg"]	10 { 11 "result": { 12 "type": "classification",	
服务调	用		}	13 "model_name": "mnist-classify", 14 "pred": 5 15 }	
公网访问	问地址 http http	o://servia os://serv		16 }	
VPC访问	问地址 http http	o://servia os://serv			
QPS	500	(单服			
接口信	息				
调用地址					
/v1/mode	els/m:predict		发送请求	取消	а.
				•	

• Sample 3. When the path {"images":["https://qcloudimg.tencent-

cloud.cn/raw/3768c5d8a47b5da666c82a286e00759b.jpeg"]} of the image of 9 is entered in the request body, the prediction result will be 9.

equest E	Body	Response	
1	<pre>{"images":["https://qcloudimg. tencent-cloud.cn/raw/ 3768c5d8a47b5da666c82a286e00759b.jpeg"] }</pre>	<pre>10 { 11 "result": { 12 "type": "classification", 13 "model_name": "mnist-classify", 14 "pred": 9 15 } 16 } </pre>	
	Send Request	est Cancel	

Demo 2. Import a third-party model into the platform and deploy it as an online service

This demo uses the PyTorch image classification model in the platform's preset model package as an example to describe how to deploy a third-party model as an online service. After deploying the model, you will be able to call the service through the online testing feature to recognize the type of the input image.

Preparations

1. Model package

(1) Log in to the TI Platform console and select **Models** > **Model Repository** on the left sidebar to enter the model list page.

(2) Click **Download Inference Code Template** in the top-right corner to get the preset model package provided by the platform.

(3) Decompress the model package and find the pytorch>classify folder.

Model Repository	Singapore 💌		Download Inference Code Template
Import Model			Enter a model name Q
Model Name	Tag ▼	Creation Time 🗘	Operation
▶ mnist_train	\bigtriangledown	2022-02-28 17:29:40	Delete Edit Label

2. Test data

The following test data has been prepared and can be directly used for call testing after the service is published.

(1) Image of cat

https://qcloudimg.tencent-cloud.cn/raw/bcbdae25439713ecb4dbb154d43a9ef8.jpeg

(2) Image of butterfly

https://qcloudimg.tencent-cloud.cn/raw/40a99b15e76d6957644f160b9149522a.jpeg

(3) Image of dog

https://qcloudimg.tencent-cloud.cn/raw/aab789b6e047fa804bbf803de16f49a0.jpeg

Step 1. Import the model repository

(1) Go to the TI Platform console and select **Models** > **Model Repository** on the left sidebar to enter the model list page.

(2) Click **Import Model** and set the following parameters in the **Import Model** pop-up:

- Import: select New model
- Model Name: customize the model name, such as classify
- Tag: add optional tags as needed
- Model Source: select COS
- Algorithm Framework: select PyTorch
- Model Metrics: enter a metric as needed, such as accuracy
- Running Environment Source: we recommend you use a built-in GPU running environment, such as py1.9.0py36-cu111
- Model File: click **Select**. In the COS pop-up window, select the target bucket, click **Upload Folder** in the bottomleft corner, upload the **pytorch>classify folder** in the model package, and select the folder path



nport Model			
Import	New model	New version	
Model Name *	classify		
	Max 60 characters; digit	; supports letters, d	iç
Tag	+ Add		
Model Source	Task OCO	S	
Algorithm Framework	PyTorch	▼	
Model Metrics	accuracy		
Running Environment Source *	Built-in / py1.9.0	-py36-cu111 🔻	
Model File 🛈 *	sophietest-1309	585087/guide/cla	
		OK	



ucket List / sophietest-130958508 / guide 💉		Search files	Q
Object Name	Size	Last Updated	
Classify/	-	-	
Jpload Files			
Jpload Files Jpload Folder			

(3) Click **OK**.

Step 2. Start an online service

(1) Go to the TI Platform console and select **Model Services** > **Online Services** on the left sidebar to enter the service list page.

(2) Click **New Service** and enter the basic service information in the pop-up window as instructed below:

- Service Name: customize the service name, such as classify
- Service Description: enter the optional service description as needed
- Billing Mode: if you have not used the resource group management module, we recommend you select Postpaid



← Start Model Service

Service Name *	classify
	Max 60 characters; supports letters, digits, underscores (_), and hyphens (-); must start with a letter or digit
Service Version	V1
Service Description (optional)	Max 500 characters
	0 / 500
Region *	Singapore
Billing Mode *	O Postpaid (pay-as-you-go)

(3) Set the instance container information as instructed below:

- Use Model File: select Yes
- Model: select the model file <code>classify</code> imported in the previous step
- Version: select v1
- Spec: select a spec as needed, such as 32C128G T4*1

Use Model File	No OYes	
Container Name	main	
Model *	classify 🔻 🗘	
Version *	v1 •	
Running Environment *	ccr.ccs.tencentyun.com/qcloud-ti-platform/ti-cloud-infer-pytorch-gpu:py3{	Select
Environment Version	py36-torch1.9.0-cu111-20211206	Version
Spec *	32C128G T4*1	
Environment Variable	+ Add	

(4) Complete the advanced configuration of the service as instructed below:

- · CLS Log Shipping: enable it as needed, which is disabled by default
- Instance Adjustment: select an option as needed, which is Manual by default
- Instance Count: configure a quantity as needed, which is 1 by default
- · Generate Authentication Token: enable it as needed, which is disabled by default
- Tag: add optional tags as needed

CLS Log Shipping 🛈	The TI console displays the logs of the last 7 days by default. If you want to store the logs persistently or use features such as log search, please enable CLS log shipping. For the CLS introduction and billing, view Document 🗹
Instance Adjustment	 Manual Set the number of instances Auto Auto scales in if the system metric value is smaller than the configured value of the trigger metric, and auto scales out if it is larger than the set value
Instance Count *	- 1 +
Generate Authentication Token	Enable
Tag 🚯	+ Add

(5) Click Start Service. If you have selected the postpaid mode, a fee freezing confirmation will be displayed. The postpaid mode requires freezing the fees for two hours in advance. If your account balance is sufficient, click OK. In the service list, the newly created service will be in Creating status. A gateway will be created, and computing resources will be scheduled in the service deployment process, which will take some time. After the service is deployed successfully, its status will change to Running.

Step 3. Test the service call

- (1) Confirm that the service published in the previous step has been deployed and is in **Running** status.
- (2) Click **Call** to enter the service call page.

Online Services	Singapore •						
New Service							Enter a name Q
Name	Status T	Billing Mode	Service ID	Running/Total Ver	Tag T	Creation Time 🕈	Operation
classify	Running	Postpaid	ms-sh5s9vxv	1/1	-	2022-02-28 20:39:	21 Add Version Edit Label Call Delete

(3) View the API information and click **Online Testing** in the API list to open the service call testing page.



API Information		
Call Address	Service Type	Operation
/v1/models/m:predict	http	Details Online Testing

(4) Enter the request information in JSON format in the request body module, click **Send Request**, and you can view the prediction result in the response module.

- Sample 1. When the path {"image": "https://qcloudimg.tencentcloud.cn/raw/bcbdae25439713ecb4dbb154d43a9ef8.jpeg"} of the image of cat is entered in the request body, the highest level of confidence will be of cat in the returned result.
- Sample 2. When the path {"image": "https://qcloudimg.tencentcloud.cn/raw/40a99b15e76d6957644f160b9149522a.jpeg"} of the image of butterfly is entered in the request body, the highest level of confidence will be of butterfly in the returned result.
- Sample 3. When the path {"image": "https://qcloudimg.tencentcloud.cn/raw/aab789b6e047fa804bbf803de16f49a0.jpeg"} of the image of dog is entered in the request body, the highest level of confidence will be of dog in the returned result.

Tencent Cloud

Getting Started with TiKit

Last updated : 2022-02-25 17:47:03

Overview

TiKit is an open-source SDK for Python provided by TI Platform, which can be used to submit training tasks to TI Platform.

Directions

Environment dependencies

Currently, Python 3.4 or later is supported.

Installing TiKit

On TI Platform, TiKit is built in the containers for notebooks and training tasks, so you don't need to install it separately.

In a non-public cloud TI Platform environment, the installation methods are 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 through pip (recommended)

```
pip3 install -U tikit
```

 Method 2. Install offline. Download the installation package at https://pypi.org/project/tikit/ and install it with the .whl file or source code:

```
pip3 install tikit-1.0.0-py3-none-any.whl
# Alternatively, after decompressing the source code, run
python3 setup.py install
```

Getting started

1. Prepare the secret_id and secret_key .

Log in to Tencent Cloud to get them on the corresponding page as shown below:

	Ticket -	Billing Center - English - 👤 -
·		541635927@qq.com Account ID: 200023315673
		 Account Information Security Settings Access Management
		Security Management RPI Access Key Preferences Dark Light
		Log Out

Cloud Access	Manage API Key					
Dashboard						
Users *	0 Safety Warning					
User Groups	• Four Ark key represents your account identity and permissions, for can operate an the reficent olocit esources under your account with reflection on Ark.					
Policies	. Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS v1.2 or above.					
Roles	• You can use the KMS white-box key to enhance API key security. For details, see KMS Best Practices 💈					
Identity Providers ~						
Access Key ^	① Usage Notes					
API Keys	• An API key is an important credential for creating TencentCloud API requests and can be used to generate a signature when you call a Tencent Cloud API 🗹. For details, see signature generation algorithms 💆.					
	disabled or deleted.					
	Create Key					
	APPID Key Creation Time Last Access Time Status Operation					
	1309211636 SecretId: IKIDJRKwIOFo68uPdbLo1xd4E7j5AyVnoY1D [] 2022-02-21 16-28-27 - On Disable					

2. Initialize the task and call the function.

from tikit.client import Client
Initialize the client. In a public cloud TI Platform environment (including not
ebooks and training tasks), you can leave the region information empty, which is
already contained in the environment variable.
client = Client("your_secret_id", "your_secret_key", "<region>")
View the list of algorithm frameworks.
client.describe_training_frameworks()

In a notebook environment, the HTML content will be directly displayed as shown below:

[10]:	<pre>client.describe_training_frameworks()</pre>					
[10]:	Framework name	Framework Version	Training mode			
	TENSORFLOW	1.15-py3.6-cpu 1.15-py3.6-cuda10.0-gpu 2.4-py3.6-cpu 2.4-py3.6-cuda11.0-gpu	PS_WORKER, MPI, HOROVOD			
	PYTORCH	1.9-py3.6-cuda11.1-gpu	DDP, MPI, HOROVOD			
	SPARK	2.4.5-cpu	SPARK			
	PYSPARK	2.4.5-py3.6-cpu	SPARK			
	LIGHT	3.1.3-py3.6-cuda11.0-gpu	DDP, MPI, HOROVOD			

In a non-notebook environment, you can print the result as shown below:

```
[12]: print(client.describe_training_frameworks())
```

Framework name	Framework Version	Training mode
TENSORFLOW	1.15-py3.6-cpu 1.15-py3.6-cuda10.0-gpu 2.4-py3.6-cpu 2.4-py3.6-cuda11.0-gpu	PS_WORKER, MPI, HOROVOD
PYTORCH SPARK PYSPARK LIGHT	1.9-py3.6-cuda11.1-gpu 2.4.5-cpu 2.4.5-py3.6-cpu 3.1.3-py3.6-cuda11.0-gpu	DDP, MPI, HOROVOD SPARK SPARK DDP, MPI, HOROVOD

3. View the function usage.

```
help(client.create_image_dataset)
```



[11]: help(client.create_image_dataset)

Help on method create_image_dataset in module tikit.client: create_image_dataset(dataset_name, storage_data_path, storage_label_path, dataset_tags=None, with_annotation=False, annotation_type=None, anno tation_format=None) method of tikit.client.Client instance Create an image dataset :param dataset_name: dataset name :type dataset name: str :param storage_data_path: COS storage path of dataset. Format: <bucket>/<cos path>/ :type storage_data_path: str

- :param storage_label_path: COS storage path of dataset label. Format: <bucket>/<cos path>/
 :type storage_label_path: str
 :param dataset_tags: list of dataset tags
- :type dataset_tags: list of tikit.tencentcloud.tione.v20211111.models.Tag :param with_annotation: Whether it has been labeled : type with_annotation: bool :param annotation_type: Labeling type. Valid values:
- ANNOTATION_TYPE_CLASSIFICATION: image classification ANNOTATION_TYPE_DETECTION: object detection ANNOTATION_TYPE_SEGMENTATION: image segmentation ANNOTATION_TYPE_TRACKING: object tracking ANNOTATION_TYPE_OCD: ANNOTATION_TYPE_OCR: OCR recognition :type annotation_type: str
- :param annotation_format: Labeling format. Valid values: ANNOTATION_FORMAT_TI: TI platform format TI platform format ANNOTATION_FORMAT_PASCAL: Pascal Voc ANNOTATION_FORMAT_COCO: C0C0
- ANNOTATION_FORMAT_FILE: file directory structure
- :type annotation_format: str :rtype: :class:`tikit.tencentcloud.tione.v20211111.models.CreateDatasetResponse`