向量数据库

性能白皮书

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【商标声明】

腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

性能白皮书
开源工具 ann-benchamrk 测试
测试环境
测试方法
测试结果

性能白皮书
开源工具 ann-benchamrk 测试
测试环境

最近更新时间：2024-09-10 14:45:51

腾讯云向量数据库（Tencent Cloud VectorDB）是一款基于向量相似度搜索的数据库产品，提供高效的向量索引和快速的相似度查询服务。为了让用户更好地了解 VectorDB 的性能表现，腾讯云发布性能白皮书，详细描述 VectorDB 的性能测试环境、测试方法和性能数据等。本章节说明测试性能所需准备的环境与数据集。

数据库规格

腾讯云向量数据库所属地域为**广州**，测试实例规格如下所示。

- 节点类型：**计算型**。
- 节点规格：**P.MEDIUM**（4CPU、8GB内存）、**P.LARGE**（8CPU、16GB内存）
- 节点数量：**3节点**。

该规格所支持的最大向量规模，请参见 产品规格 。请按照如上规格购买并新建向量数据库，具体操作，请参见 新建数据库实例 。

客户端测试环境

与腾讯云向量数据库实例为同一地域同一 VPC 内的 Linux 云服务器 CVM ，其环境要求如下：

- 操作系统：TencentOS Server 3.1 (TK4)
- 规格：SA3.16XLARGE256。
- 已安装 Python 3.6.8 及以上版本。

> **说明：**
> 使用腾讯云 CVM 连接腾讯云向量数据库（Tencent Cloud VectorDB），在腾讯云 CVM 安全组中需配置**出站规则**，把腾讯云向量数据库的 IP 及端口添加到出站规则中。在腾讯云向量数据库安全组中配置**入站规则**，把 CVM 的 IP 地址及向量数据库的端口添加到入站规则中，才能连接成功。具体操作，请参见 安全组 。

测试工具

开源工具 ann-benchamrk 是一个用于评估近似最近邻（ANN）搜索库的性能测试工具。它提供了一套标准的测试数据集和评估指标，可以用于比较不同量级数据库的性能表现。腾讯云向量数据库基于 ann-benchmark 进行数据库性能评测，与竞品公平的进行性能差异对比。如下基于运行工具的命令，介绍其关键参数。

```
python3 run.py --dataset sift-128-euclidean  --local --force --algorithm
vector_db --definitions=configs/8c64g/100w-1shard.yml --batch --
only_query --runs 1 -k 10
```

参数	参数含义
--dataset	指定数据集的名字，数据集放在 ./data 目录下。
--local	指定是否在本地运行。
--force	运行查询测试时，如果历史已经测试过（存在本地测试结果 ），是否继续执行并覆盖。
--algorithm	ann-benchmark 默认支持多种数据库的测试，这里强制指定测试数据库为 vector_db。
--definitions	指定测试运行时的配置文件。
--batch	添加该参数后，测试工具会使用 CPU 多核模式压测数据查询，使用的 CPU 核数可以通过配置文件中的 Threads 参数控制。
--runs	指定查询运行的次数。当运行性能测试时，期望工具能长时间运行时，可将该参数设置为较大值，如999999。
--only_query	指定是否只运行查询。 因 ann-benchmark 默认的工作方式是先插入数据，再运行查询测试。 但在实际使用中，我们可能存在插入一次数据后，多次进行查询测试的场景。增加该参数后，可以跳过 ann-benchmark 的数据插入阶段。
-k	希望查询返回的数据条数。

数据集

ann-benchamrk 官方数据集无需提前下载，测试工具运行时会自动检查 ./data（ ann-benchamrk 工具存放数据集的路径 ）目录下是否存在数据集文件，如果不存在则会主动连接官网下载官方数据集。ann-benchamrk 官方不具备768维度的数据集，腾讯云向量数据库团队基于中文文本制作了768维度，从100W 到1000W 级别的数据集供测试选用。具体数据集信息，如下表所示。

数据集名	数据集介绍	向量维度	向量数	索引方法	距离类型
sift-128-euclideam	官方数据集	128	1,000,000	HNSW	L2
gist-960-euclidean	官方数据集	960	1,000,000	HNSW	L2

chinese100w-768-angular	腾讯云向量数据库提供的数据集	768	1,000,000	HNSW	IP
chinese500w-768-angular	腾讯云向量数据库提供的数据集	768	5,000,000	HNSW	IP
chinese1000w-768-angular	腾讯云向量数据库提供的数据集	768	10,000,000	HNSW	IP

版权所有：腾讯云计算（北京）有限责任公司

测试方法

最近更新时间：2024-09-09 12:15:50

腾讯云向量数据库（Tencent Cloud VectorDB）采用开源工具 ann-benchmark 进行性能测试，并且为适应云上测试场景，对 ann-benchmark 工具进行了改造适配。本文详细介绍基于 ann-benchmark 工具进行数据库性能测试的方法。

下载并上传工具与数据集

1. 下载测试工具。

   ann-benchmarks-dev.zip
2. 获取腾讯云向量数据库提供的数据集。

> **说明：**
> ann-benchamrk 官方数据集默认不需要提前下载，测试工具运行时会自动检查 ./data 目录下是否存在数据集文件，如果不存在则会主动联网下载。

3. 登录云服务器 CVM 环境，使用 `rz` 命令上传测试工具于 CVM。CVM 要求，请参见 测试环境 。
4. 执行 `unzip ann-benchmarks-dev.zip` 命令解压测试工具压缩包 ann-benchmarks-dev.zip 。
5. 使用 `rz` 命令上传数据集于测试工具解压目录 `../ann-benchmarks-dev/ann-benchmarks/data` 。

测试 128 维数据在 HNSW 索引下单核查询性能

步骤1：安装工具环境依赖

进入已解压的测试工具包的路径，安装 python 运行依赖。

```
cd ann-benchmarks
pip3 install -r requirements.txt
```

步骤2：修改配置文件

执行如下命令，拷贝配置文件，并打开配置文件，配置相关参数。需配置的参数信息，请参见下表，其余参数保持默认值即可。

```
cp ann_benchmarks/algorithms/vector_db/config.yml mytest.yml
vi mytest.yml
```

配置文件参数	参数含义	建议值

arg_groups 下的第一个参数	索引类型为 HNSW 的参数 M，指每个节点在检索构图中可以连接多少个邻居节点	设置 16
HttpBase	连接向量数据库访问地址	获取向量数据库实例的内网 IP 地址与网络端口。具体操作，请参见 查看实例信息 。
User	连接用户名	root
ApiKey	向量数据库的 API 访问密钥	如何获取，请参见 密钥管理 。
IndexType	索引类型	HNSW
MetricType	相似度计算方法	L2
ColReplicaNum	副本数	2
Threads	单线程进行测试	1
ColShardNum	分片数量	3
EfConstruction	索引类型为 HNSW 的参数 ef，搜索时，指定寻找节点邻居遍历的范围	500
BuildIndexOnUpsert	指定插入数据时，同步更新索引	• true：重新创建索引。 • false：不重新创建索引。
query_args	指定了在数据插入完成后，需要进行ef为10，15，20 等的查询	[10, 15, 20, 25, 30, 35, 40, 50, 60, 80, 120, 200, 400]

```
float:
  any:
    - base_args: [ '@metric' ]
      constructor: VectorDb
      disabled: false
      docker_tag: ann-benchmarks-vector_db
      module: ann_benchmarks.algorithms.vector_db
```

```yaml
    name: vector_db
    run_groups:
      vector_db:
        arg_groups:
          - [ 16 ]
          - DbName: db-test
            HttpBase: http://127.0.0.1:18100
            NeedAuth: true
            User: root
            ApiKey: invalid-by-default
            DropDb: false
            CreateDb: true
            ReCreateCollection: false
            DropCollectionOnDone: false
            IndexType: HNSW
            MetricType: L2
            ColReplicaNum: 2
            ColShardNum:
            EfConstruction: 500
            ExitOnError: true
            Threads: 1
            MultiCpuUpsert: true
            UpsertBatchSize: 500
            BuildIndexOnUpsert: true
            KNNSeekMode: false
            KNNSeeKStartEf: 90
            KNNSeekStep: 1
            KNNSeekExpect: 0.95
        args: { }
        query_args:
          - [ 10, 15, 20, 25, 30, 35, 40, 50, 60, 80, 120, 200, 400 ]
          - RetrieveVector: false
```

步骤3：运行测试工具

执行如下命令，运行测试工具。其中，--dataset 指明数据集名称。 vector_db 指明了数据库名，数据库名从 mytest.yml 中获取。具体参数含义，请参见 测试工具 。

> **说明：**
> euclidean 使用的是 L2 相似度算法，angular 使用的是 IP 算法。

```
python3 run.py --dataset sift-128-euclidean  --local --force --
parallelism 1 --algorithm vector_db --definitions=mytest.yml --runs 1
```

步骤4：查看测试运行结果

测试结束后，测试结果保存在 results 文件夹中，需要使用如下命令将结果转换为可读的 csv 文件。如下所示，可通过生成的 mytest.csv 文件，查看 128 维数据在 HNSW 索引下单核的性能表现。

```
python3 data_export.py --output=mytest.csv
```

探索指定召回率，需设置的查询 EF 值

步骤1：打开探索模式

打开配置文件 mytest.yml，修改配置文件中的 **KNNSeekMode** 参数为 **true**， 该模式测试工具会反复运行不同 ef 值的查询，直到获得最匹配的召回率为止。

```
31            BuildIndexOnUpsert: true
32            KNNSeekMode: false
33            KNNSeeKStartEf: 90
34            KNNSeekStep: 1
35            KNNSeekExpect: 0.95
36        args: { }
```

步骤2：配置探索参数

配置文件参数	参数含义	建议值
KNNSeekStartEF	指定开始查询 ef 设置的起始值，即从哪一个ef 参考值开始查询	90
KNNSeekStep	指定探索模式中，每次 ef 值变化几个单位。如1，则表示 KNNSeekStartEF+=1 或相减，每次递增或递减的步长	1
KNNSeekExpect	期望找到的召回率	0.95

步骤3：运行工具

具体参数含义，请参见 测试工具 。

```
python3 run.py --dataset sift-128-euclidean  --local --force --
parallelism 1 --algorithm vector_db --definitions=mytest.yml --runs 1 --
```

```
only_query
```

步骤4： 查看运行结果

结果显示如下图所示，KNNSeekExpect=0.95，即在ef=61时，可获得最接近0.95的召回率。

```
Processed Query, TopK=10 : 100%|                                                                                    | 3000/3000 [00:03<00:00, 887.60it/s]
==== Seek for ef: 61, get knn: 0.9493
All seek results: {'70': 0.9609, '69': 0.9599333333333334, '68': 0.9589333333333334, '67': 0.9577666666666668, '66': 0.9566666666666667, '65': 0.9553666666666667, '64': 0.9540333333333333, '63': 0.9526666666666668,
'62': 0.9512666666666666, '61': 0.9493}
Best match ef: 61, matched KNN: 0.9493
2023-08-03 20:57:54.597 - annb - INFO - Terminating 1 workers
[root@VM-4-12-tencentos ann-benchmarks]#
default
```

Search 检索查询性能

步骤1：设置期望的 EF 值，压测查询配置

在配置文件中，如下使用 8 核压测查询 `ef=111` 的情况。注意需要设置 KNNSeekMode为 false。

```
27          ExitOnError: true
28          Threads: 8
29          MultiCpuUpsert: true
30          UpsertBatchSize: 500
31          BuildIndexOnUpsert: true
32          KNNSeekMode: false
33          KNNSeekStartEf: 90
34          KNNSeekStep: 1
35          KNNSeekExpect: 0.95
36        args: { }
37        query_args:
38          - [ 111 ]
39          - RetrieveVector: false
40
```

步骤2：运行工具

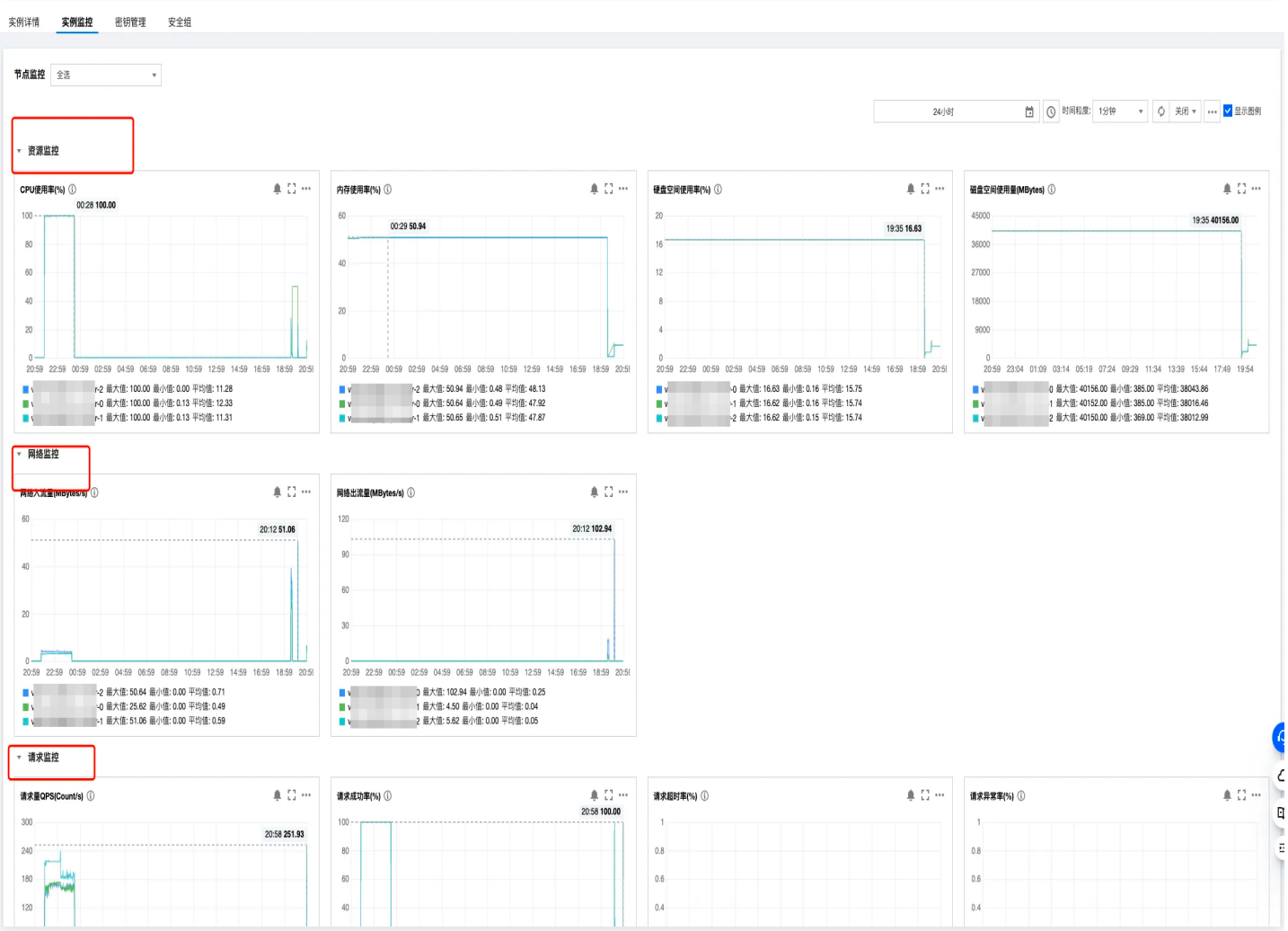
压测命令，压测时需要打开--batch 参数，同时设置--runs 为较大值以便长时间运行。具体参数含义，请参见 测试工具 。

```
python3 run.py --dataset sift-128-euclidean  --local --force --
parallelism 1 --algorithm vector_db --definitions=mytest.yml --runs 999
```

```
--only_query --batch
```

步骤3：查看结果

腾讯云向量数据库控制台提供了实例的 CPU，内存、QPS，时延等关键性能指标监控。具体操作，请参见 查看监控数据 。



说明：

受限于 python 语言的线程模型，有时测试工具无法把测试压力打满，导致实例性能无法达到极限，此时可使用多进程或多台 CVM 同时进行压测。

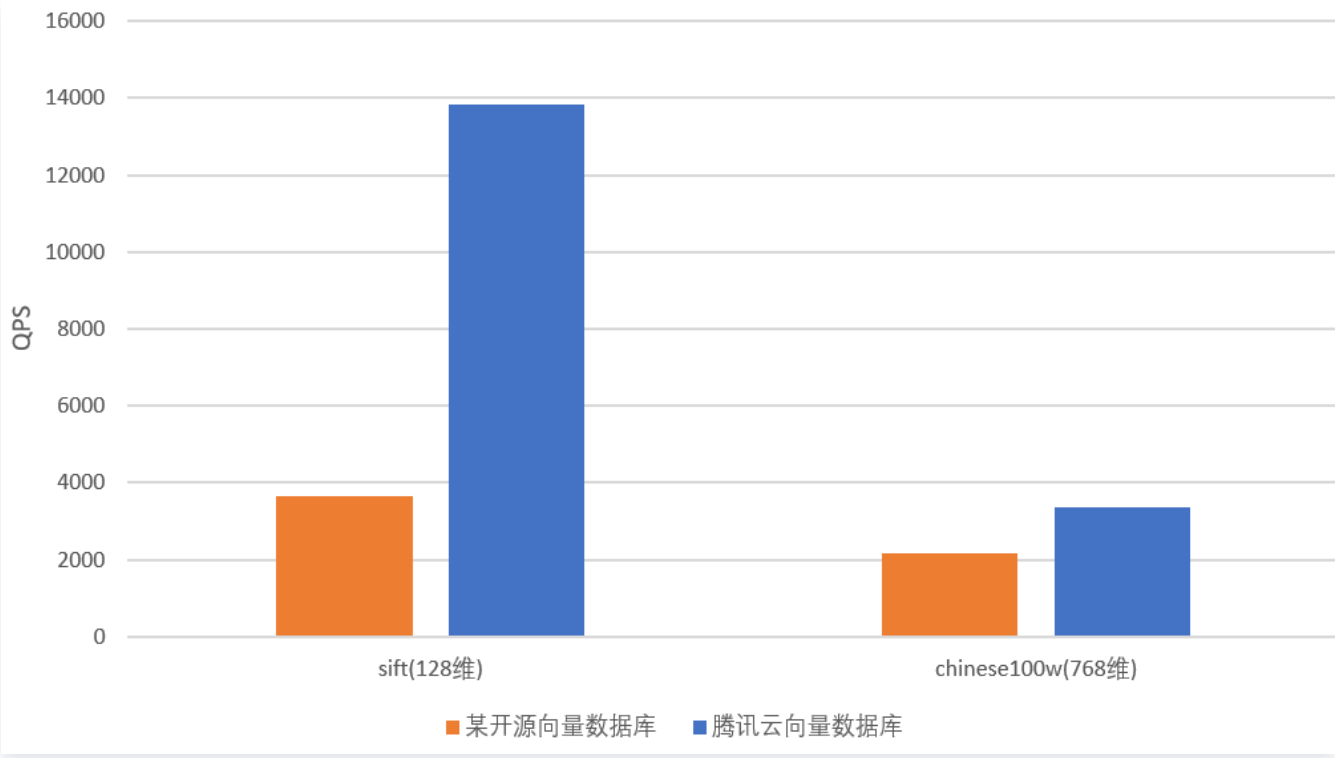
测试结果

最近更新时间：2024-09-09 12:15:50

本文提供了腾讯云向量数据库（Tencent Cloud VectorDB）采用开源工具 ann-benchmark 进行性能测试的详细数据。吞吐量 QPS 是指系统在单位时间内能够处理的查询请求数量，是衡量系统查询处理能力的重要指标。该性能测试集中测试不同维度的 QPS 数据、不同召回率下的 QPS 数据、不同数据规模的 QPS 数据。

不同维度最大 QPS 对比

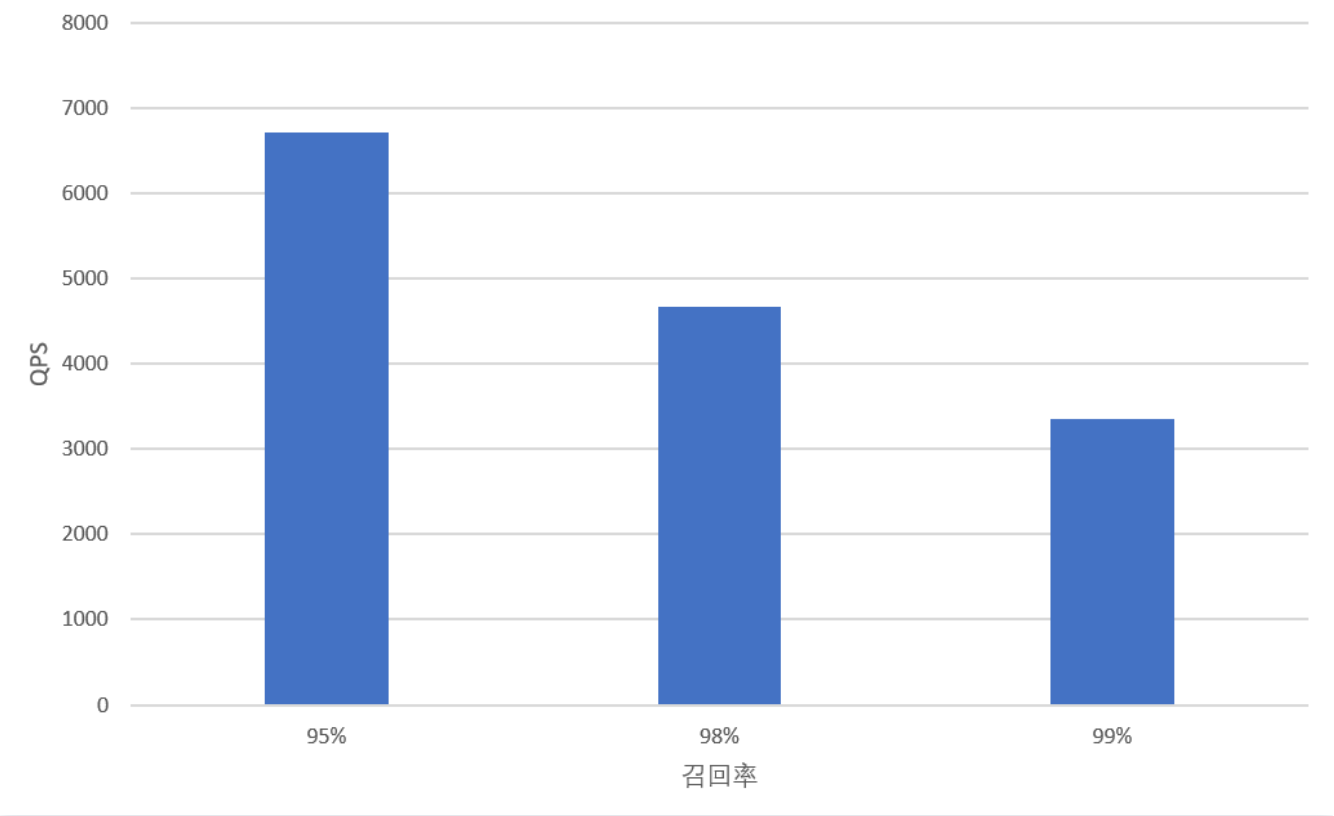
- **测试目标**：检索不同维度的数据集，向量索引选择 HNSW 类型，在召回率达到 99% 的情况下，获取最相似的Top10的文档，对比某开源向量数据库与腾讯云向量数据库的 QPS 数据。
- **测试规格**：向量数据库 **P.MEDIUM**（4CPU、8GB内存）、节点数量为 3。
- **测试数据集**：数据量级100w，128、768、960三档维度的数据集
- **测试结论**：数据集128维与768维某开源向量数据库与腾讯云向量数据库的 QPS 对比测试数据，如下所示。通过如下对比视图，可看出腾讯云的 QPS 性能具有显著优势。通过该项测试，可得出如下结论：
  - 在不同维度的数据集下，HNSW 索引都可以达到99%以上的召回率。
  - 在数据量相同的情况下，随着向量维度的增加，检索时资源开销增加，腾讯云向量数据库 QPS 会有所降低。
  - 同一数据集，与某开源自建向量数据库对比，腾讯云向量数据库的 QPS 有36%到279%的提升。对比视图，如下所示。



数据集	分片和副本	索引类型	召回率	QPS	
				某开源向量数据库	腾讯云向量数据库
sift-128-euclideam	shardNum=1 replicaNum=2	HNSW（m=16,efConstruction=200）	99%	3653	13843 (↑279%)
chinese100w-768-angular				2166	3346 (↑54%)
gist-960-euclidean				480	651 (↑36%)

不同召回率下的 QPS 对比

- **测试目标**：检索100w数据量级，向量索引类型为 HNSW，设置不同的查询参数 ef（指定寻找节点邻居遍历的范围），分析召回率的变化，对比 QPS 数据。
- **测试规格**：向量数据库 P.MEDIUM（4CPU、8GB内存）、节点数量为 3。
- **测试数据集**：chinese100w-768-angular，数据量级100w，768维度的数据集。
- **测试结论**：测试数据，如下表所示。分析可得出如下结论。
  - 同一数据集，召回率要求越高，即 ef 参数设置越大时，QPS越低。 不同召回率 QPS 的对比视图，如下所示。
  - 同一数据集，在其他配置不变的情况下，若需提高召回率，可适当增加查询参数 ef。

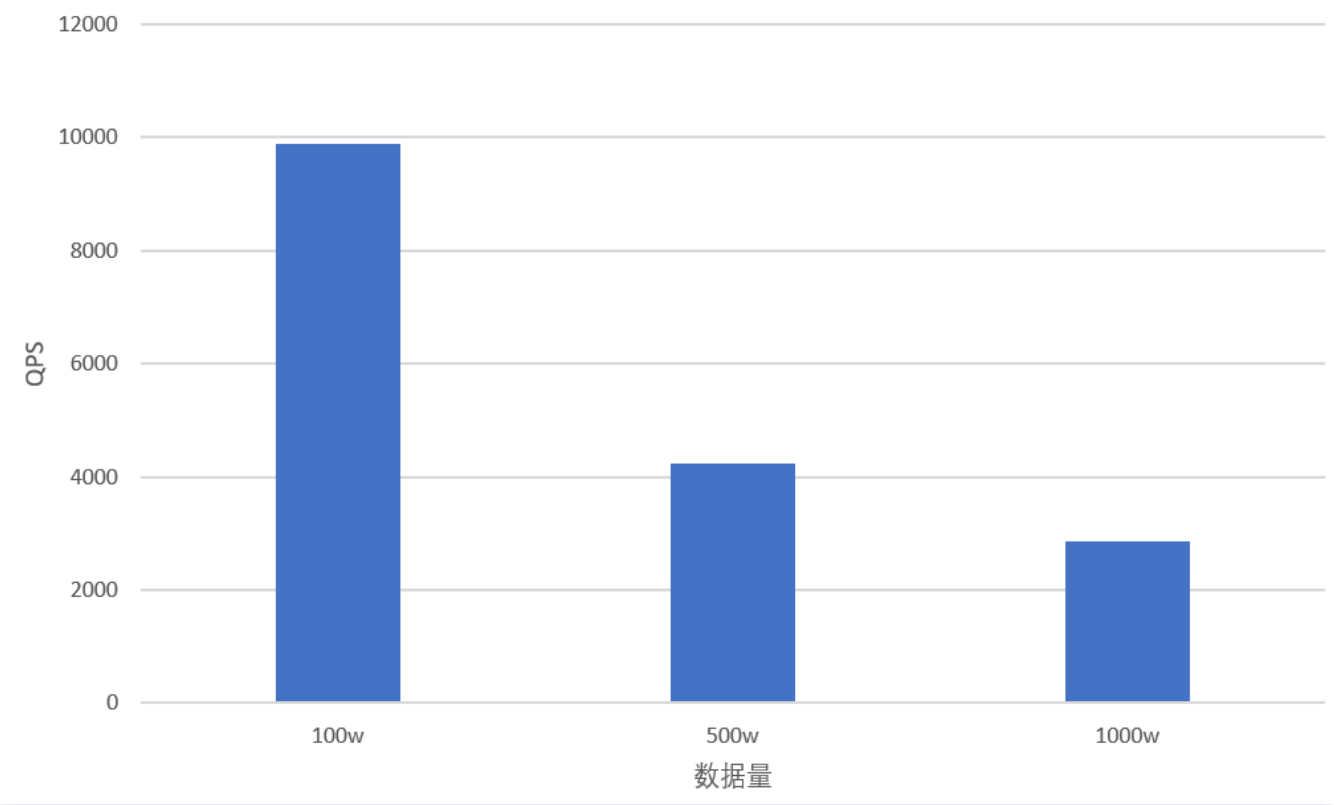


数据集	分片和副本	索引类型	查询参数	召回率	QPS
chinese100w-768-angular	shardNum=1 replicaNum=2	HSNW（m=16,efConstruction=200）	ef=85	95%	6708
			ef=149	98%	4671
			ef=200	99%	3346

不同数据规模 QPS 对比

- **测试目标：**检索不同数据规模的数据集，在召回率达到95%的情况下，获取最相似的 Top10的文档，分析不同数据量级的 QPS 数据。
- **测试规格：**向量数据库 **P.LARGE**（8CPU、16GB内存）、节点数量为 3。
- **测试数据集：**chinese100w-768-angular、chinese500w-768-angular、chinese1000w-768-angular 768维度不同数据量级的数据集。
- **测试结论：**测试数据，如下表所示。分析可得出如下结论。
  - 向量维度相同时，随着数据规模量级增加，检索时资源开销增加，QPS 有所降低。具体对比视图，如下所示。
  - 从数据中可以看出，数据量越大，达到相同召回率时，需要设置的查询参数 ef 就会越大，检索过程中便需要遍历更大的搜索范围。因此，在实际应用HNSW 索引时，需要根据业务规模和召回率要求，提前测试并设

置合理的 ef 参数。



数据集	数据量	分片和副本	索引类型	查询参数	召回率	QPS
chinese100w-768-angular	1,000,000	shardNum=1 replicaNum=2	HNSW（m=16,efConstruction=500）	ef=112	95%	9891
chinese500w-768-angular	5,000,000		HNSW（m=16,efConstruction=500）	ef=193		4223
chinese1000w-768-angular	10,000,000		HNSW（m=16,efConstruction=500）	ef=224		2864