

原创

single-coder

于 2018-11-20 12:34:33 发布

5599

收藏 4


版权

分类专栏：

深度学习

文章标签：

深度学习

深度学习 专栏收录该内容

0 订阅

7 篇文章

订阅专栏

选择合适的index类型

选择index类型并没有一套精准的法则可以依据，需要根据自己的实际情况选取。下面的几个问题可以作为选取index的参考。

是否需要精确的结果

如果需要，应该使用"Flat" 只有 IndexFlatL2 能确保返回精确结果。一般将其作为baseline与其他索引方式对比，以便在精度和时间开销之间做权衡。

不支持add_with_ids，如果需要，可以用"IDMap, Flat"。

支持GPU。

In [8]:

```
#导入faiss
import sys
sys.path.append('/home/maliqi/faiss/python/')
import faiss

#数据
import numpy as np
d = 512 #维数
n_data = 2000
np.random.seed(0)
data = []
mu = 3
sigma = 0.1
for i in range(n_data):
    data.append(np.random.normal(mu, sigma, d))
data = np.array(data).astype('float32')

#ids, 6位随机数
ids = []
start = 100000
for i in range(data.shape[0]):
    ids.append(start)
    start += 100
ids = np.array(ids)
```

In [9]:

```
#不支持add_with_ids
index = faiss.index_factory(d, "Flat")
index.add(data)
dis, ind = index.search(data[:5], 10)
print(ind)
```

```
[[ 0  798  879  223  981 1401 1458 1174  919  26]
 [ 1  981 1524 1639 1949 1472 1162  923  840  300]]
```

目录

选择合适的index类型

是否需要精确的结果

关心内存开销

如果不在意内存占用空间，使用...

如果稍微有点在意，使用"..., Flat"

如果很在意，使用"PCARx...,SQ8"

如果非常非常在意，使用"OPQx...

数据集的大小

如果小于1M，使用"...,IVFx,..."

如果在1M-10M，使用"...,IMI2x1..."

如果在10M-100M，使用"...,IMI2...

分类专栏

Hand-on deep learnin...

deep learning

剑指offer Python版 9篇

深度学习 7篇



```
[ 2 1886 375 1351 518 1735 1551 1958 390 1695]
[ 3 1459 331 389 655 1943 1483 1723 1672 1859]
[ 4 13 715 1470 608 459 888 850 1080 1654]]
```

In [10]:

```
index = faiss.index_factory(d, "IDMap, Flat")
index.add_with_ids(data, ids)
dis, ind = index.search(data[:5], 10)
print(ind) # 返回的结果是我们自己定义
```

```
[[100000 179800 187900 122300 198100 240100 245800 217400 191900 102600]
 [100100 198100 252400 263900 294900 247200 216200 192300 184000 130000]
 [100200 288600 137500 235100 151800 273500 255100 295800 139000 269500]
 [100300 245900 133100 138900 165500 294300 248300 272300 267200 285900]
 [100400 101300 171500 247000 160800 145900 188800 185000 208000 265400]]
```

关心内存开销

需要注意的是faiss在索引时必须将index读入内存。

如果不在意内存占用空间，使用 “HNSWx”

如果内存空间很大，数据库很小，HNSW是最好的选择，速度快，精度高，一般 $4 \leq x \leq 64$ 。不支持add_with_ids，不支持移除向量，不需要训练，不支持GPU。

In [12]:

```
index = faiss.index_factory(d, "HNSW8")
index.add(data)
dis, ind = index.search(data[:5], 10)
print(ind)
```

```
[[ 879 981 26 1132 807 1639 28 1334 1832 1821]
 [ 1 981 1524 1639 1949 1472 1162 923 840 300]
 [ 2 1886 375 1351 518 1958 390 1695 1707 1080]
 [ 3 1459 331 389 655 1483 1723 1672 1859 650]
 [ 4 13 715 1470 608 459 1080 1654 665 154]]
```

如果稍微有点在意，使用 “..., Flat”

"..."是聚类操作，聚类之后将每个向量映射到相应的bucket。该索引类型并不会保存压缩之后的数据，而是保存原始数据，所以内存开销与原始数据一致。通过nprobe参数控制速度/精度。支持GPU,但是要注意，选用的聚类操作必须也支持。

In [14]:

```
index = faiss.index_factory(d, "IVF100, Flat")
index.train(data)
index.add(data)
dis, ind = index.search(data[:5], 10)
print(ind)
```

```
[[ 0 879 981 1401 919 143 2 807 1515 1393]
 [ 1 511 1504 987 747 422 1911 638 851 1198]
 [ 2 879 807 981 1401 1143 733 441 1324 1280]
 [ 3 740 155 1337 1578 1181 1743 290 588 1340]]
```

```
[ 4 1176 256 1186 574 1459 218 480 1828 942]]
```

如果很在意，使用”PCARx,...,SQ8 “

如果保存全部原始数据的开销太大，可以用这个索引方式。包含三个部分，

- 1.降维
- 2.聚类
- 3.scalar 量化，每个向量编码为8bit 不支持GPU

In [19]:

```
index = faiss.index_factory(d, "PCAR16,IVF50,SQ8") #每个向量降为16维
index.train(data)
index.add(data)
dis, ind = index.search(data[:5], 10)
print(ind)
```

```
[[ 0 671 196 1025 624 1521 724 879 1281 533]
 [ 1 1008 698 206 101 657 294 383 700 574]
 [ 2 1594 754 1850 266 559 154 1723 1949 1910]
 [ 3 1758 820 869 1067 14 211 1214 78 1445]
 [ 4 1457 466 557 1604 1951 912 736 1974 836]]
```

如果非常非常在意，使用”OPQx_y,...,PQx“

y需要是x的倍数，一般保持y<=d，y<=4*x。支持GPU。

In [26]:

```
index = faiss.index_factory(d, "OPQ32_512,IVF50,PQ32")
index.train(data)
index.add(data)
dis, ind = index.search(data[:5], 10)
print(ind)
```

```
[[ 0 1686 1186 1552 47 517 1563 1738 1748 125]
 [ 1 747 1816 41 1599 380 1179 803 1964 422]
 [ 2 1610 1886 928 397 874 676 535 1401 929]
 [ 3 548 89 509 1337 865 1472 1210 1181 1578]
 [ 4 260 1781 1001 1179 41 20 747 1803 1055]]
```

数据集的大小

在高效检索的index中，聚类是其中的基础操作，数据量的大小主要影响聚类过程。

如果小于1M，使用”...,IVFx,...“

N是数据集中向量个数，x一般取 $\lceil 4\sqrt{N} \rceil, \lceil 16\sqrt{N} \rceil$,需要 $30x \sim 256x$ 个向量的数据集去训练。

如果在1M-10M，使用”...,IMI2x10,...“

使用k-means将训练集聚类为 2^{10} 个类，但是执行过程是在数据集的两半部分独立执行，即聚类中心有 $2^{(2^{10})}$ 个。

如果在10M-100M，使用”...,IMI2x12,...“

single-coder

码龄4年

暂无认证

18

21万+

102万+

7万+

原创

周排名

总排名

访问

等级

770

9

23

33

107

积分

粉丝

获赞

评论

收藏

私信

关注

搜博主文章

- 最新评论
- faiss安装教程

If you bloom,butterflies ll come: win没有gpu的版本
- 视频分类数据集介绍

我那21克的灵魂: 点一个大大的赞
- faiss安装教程

m0_37905285: 这个是链接https://github.com/facebookresearch/faiss
- faiss安装教程

m0_37905285: 因为你需要去github上下载,这个包是网上链接没有的。
- 视频分类数据集介绍

百里不守约_45690024: http://crcv.ucf.edu/data/UCF101/UCF101.rar 你好,这个网...

深度学习之以图搜图实战 (PyTorch + Faiss)	10-26
<p class="ql-long-32928933" style="line-height: 1.7; margin-bottom: 0pt; margin-top: 0pt; font-size: 11pt; color: #494949;"><span class="ql-author-329...	
参与评论 您还未登录, 请先 登录 后发表或查看评论	
Faiss框架学习_u012477435的博客_faiss框架	2-11
3.2 indexBinary 3.2.1 IndexBinary创建 IndexBinaryFlat IndexBinaryIVF IndexBinaryHNSW faiss.index_binary_factory(d,"BIVF32") 3.2.2 IndexBinary方法 ...	
Faiss教程:入门_nuohanfengyun的博客_faiss search	2-8
index= faiss.index_factory(d,"IVF100,PQ8") PQ8替换为Flat便得到了IndexFlat索引,工厂方法是非常有效的,尤其是对数据采用预处理的时候,如参数PCA3...	
【Faiss】基础索引类型 (六)	mjiansun的专栏 609
基础索引类型 数据准备 import numpy as np d = 512 #维数 n_data = 2000 np.random.seed(0) data = [] mu = 3 sigma = 0.1 for i in range(n_data): data...	
faiss-index进阶操作	dake1994的博客 5659
index进阶操作 下面介绍的方法只支持部分Index类型。 从index中恢复出原始数据 给定id, 可以使用reconstruct或者reconstruct_n方法从index中回复出原...	
【Faiss】索引选择指南(三)_mjiansun的专栏	1-18
index = faiss.index_factory(d,"Flat") index.add(data) dis, ind = index.search(data[:5],10) print(ind) [[079887922398114011458117491926] [198115241...	
Faiss从入门到实战精通_bitcarmanlee的博客_faiss 入门	2-13
2. 为数据集选择合适的index,index是整个faiss的核心部分,将第一步得到的训练数据add到index当中。 3.search,或者说query,搜索到最终结果。 4.faiss原...	
Faiss教程：基础	weixin_33748818的博客 662
Faiss对一些基础算法提供了非常高效的实现：k-means、PCA、PQ编解码。 聚类 假设2维tensor x： ncentroids = 1024 niter = 20 verbose = True d = x...	
一文看懂HNSW算法理论的来龙去脉	u011233351的博客 4万+
HNSW算法---Hierarchical Navigable Small World graphs，第一贡献者：Y.Malkov(俄) 一.背景介绍 在浩渺的数据长河中做高效率相似性查找一直以...	
Faiss教程:基础_nuohanfengyun的博客	2-11
推荐使用index_factory,通过参数创建索引。 Flat 提供数据集的基准结果,不压缩向量,也不支持添加id,如果需要 add_with_ids,使用"IDMap,Flat"参数。 无需...	
faiss IndexFlat源码详解	168
IndexFlat原理比较简单,把add进来的原始向量,全部保存起来。 在检索时, query向量和索引中的所有的原始向量求距离 这种暴力批量的方式,性能非...	
faiss hnsW 算法源码详解 - train	716
hnsWlib 代码分析hnsWlib 源码分析 train过程说明 主要是生成hnsW模型 HnsW中, storage中存储的原始的中心点向量 生成hnsW 为每层分配空间 每层的中...	
【Faiss】简介及示例,索引类型	mjiansun的专栏 1843
https://blog.csdn.net/kanbuqinghuanyizhang/article/details/80774609	
faiss技术积累	杨树的博客 9132
Faiss教程：入门 https://www.cnblogs.com/houkai/p/9316129.html Faiss教程：基础 https://www.cnblogs.com/houkai/p/9316136.html Fais...	
【Faiss】indexes 前(后)处理 (五)	mjiansun的专栏 2100
Pre and post processsing 在某些情形下,需要对Index做前处理或后处理。 ID映射 默认情况下, faiss会为每个输入的向量记录一个次序id,在使用中也可...	
faiss简介及示例 热门推荐	JC的博客 4万+
简介 faiss是为稠密向量提供高效相似度搜索和聚类的框架。由Facebook AI Research研发。 具有以下特性。 1、提供多种检索方法 2、速度快 3、可存在...	
Faiss教程：入门	weixin_34389926的博客 807
Faiss处理固定维度d的数据, 矩阵每一行表示一个向量, 每列表示向量的一项。Faiss采用32-bit浮点型存储。 假设xb为数据集, 维度为nb × d ; xq是查...	
HNSW算法原理 (一)	CHIERYU的专栏 7751
原文链接：https://blog.csdn.net/CHIERYU/article/details/81989920 HNSW算法可类比于skip lists数据结构,对于增和查操作,其与skip lists有很多相同之...	
faiss入门+使用的索引原理	zlb872551601的博客 2404
faiss入门+使用的索引原理	
faiss 三种基础索引方式 最新发布	xian0710830114的专栏 219
faiss 三个最基础的 index. 分别是IndexFlatL2,IndexIVFFlat,IndexIVFPQ 一、 IndexFlatL2 - 最基础的Index IndexFlatL2索引的结果是精确的,可以用来作为...	
Faiss建立索引并保存 (C++)	jiehanwang的专栏 1132
Faiss 建立索引并保存。如果用IndexHNSWFlat,就采用IndexIDMap进行映射。 #include "index_io.h" // #include "IndexIVF.h" // #include "IndexIVFFlat.h" #...	

您愿意向朋友推荐“博客详情页”吗？

强烈不推荐不推荐一般般推荐强烈推荐

最新文章

大疆创新笔试(2019-08-04)

git报错 warning: Clone succeeded, but checkout failed.

faiss-index进阶操作

2019年 2篇

2018年 16篇

Faiss教程：索引(1)

weixin_33872660的博客 1026

索引是faiss的关键知识，我们重点介绍下。索引方法汇总 有些索引名，我就不翻译了，根据英文名去学习更准确。索引名 类名 index_factory 主要参数 ...

对Faiss中IndexFlatL2、IndexIVFFlat、IndexIVFPQ三种索引的总结和选择

qysh123的专栏 548

由于项目和研究的需要，想要存储并检索大量的embedding，在之前的博客里，我尝试了一种方案：https://blog.csdn.net/qysh123/article/details/113754...

IndexFlatL2、IndexIVFFlat、IndexIVFPQ三种索引方式示例

weixin_30396699的博客 1449

上文针对Faiss安装和一些原理做了简单说明，本文针对标题所列三种索引方式进行编码验证。首先生成数据集，这里采用100万条数据，每条50...

【Faiss】indexes IO和index factory (四)

mjiansun的专栏 463

I/O操作 faiss.write_index(index, "index_file.index") #将index保存为index_file.index文件 index = faiss.read_index("index_file.index") #读入index_file.index...

©2022 CSDN 皮肤主题：深蓝海洋 设计师：CSDN官方博客 返回首页

关于我们 招贤纳士 商务合作 寻求报道 400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

公安备案号11010502030143 京ICP备19004658号 京网文〔2020〕1039-165号 经营性网站备案信息 北京互联网违法和不良信息举报中心 家长监护 网络110报警服务 中国互联网举报中心 Chrome商店下载 ©1999-2022北京创新乐知网络技术有限公司 版权与免责声明 版权申诉 出版物许可证 营业执照