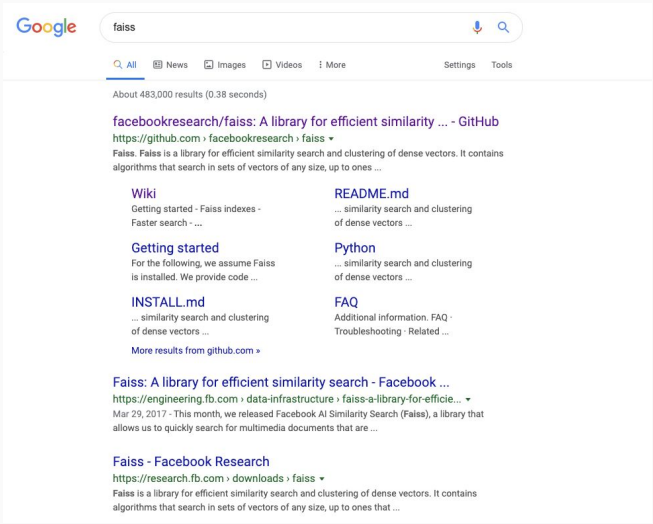
Faiss索引简介

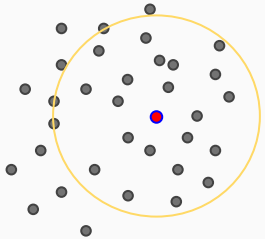
廖长增
liaochangzeng@zhihu.com

问题背景



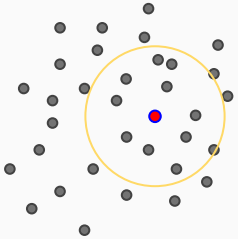
检索

推荐



海量数据计算K近邻

暴力法



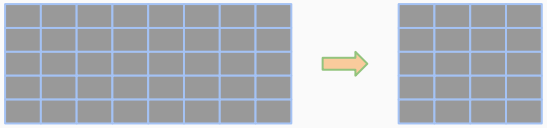
海量数据计算K近邻

对于每一次查询计算库内所有向量的相似性，返回最相似的 K个向量

- 能返回精确结果
- 时间复杂度为O(N)，计算耗时大
- 所有向量一起计算，会执行大量无效的计算
- 只适用于极小规模的数据集

暴力法的优化

压缩编码



计算向量的压缩编码

将原始向量压缩表示为维度较低的向量

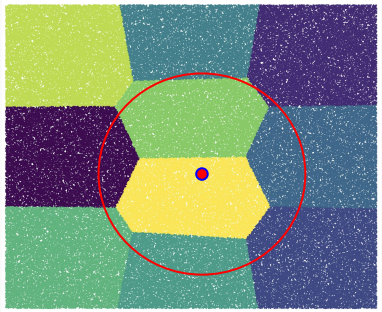
- 速度更快，但只能得到近似 结果
- 检索结果的精度依赖于编码的粒度
- 所有向量需要一起 计算，会执行大量无效的计算
- 无法在超大规模数据上应用

常用压缩编码方法

- PQ(Product Quantizer、乘积量化)
- LSH(Local Sensitive Hash、局部敏感哈希)
- PCA(Principal Component Analysis、主成分分析)

暴力法的优化

限制搜索空间(HNSW、IVF)



对搜索空间进行预先划分

对搜索空间进行预划分，搜索时只在相邻的块内召回候选集

- 无需计算所有向量的相似性，极大 缩减计算量
- 结合压缩编码可在超大规模数据上应用

常用方法

- IVF(Inverted File、倒排索引)
- HNSW

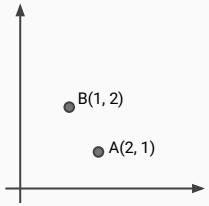
Faiss介绍

Faiss是Facebook AI团队开源的针对聚类和相似性搜索库，为稠密向量提供高效相似度搜索和聚类，支持十亿级别向量的搜索，是目前最为成熟的近似近邻搜索库。使用C++编写，提供完美与numpy完美衔接的python接口。百万数据集上能做到毫秒级的延迟，在万亿数据集上能做到分钟级的延迟。

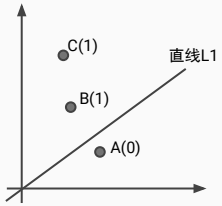
优化点：

1. 提供高效的压缩编码实现方式(PQ、LSH、SQ)
2. 实现了业界先进的索引算法(HNSW)
3. 做了大量工程上的并行性优化(多线程、GPU计算)
4. 在基础索引的基础上支持复杂的组合索引

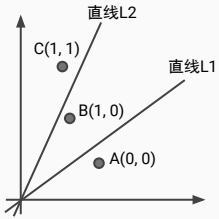
LSH(Local Sensitive Hash)



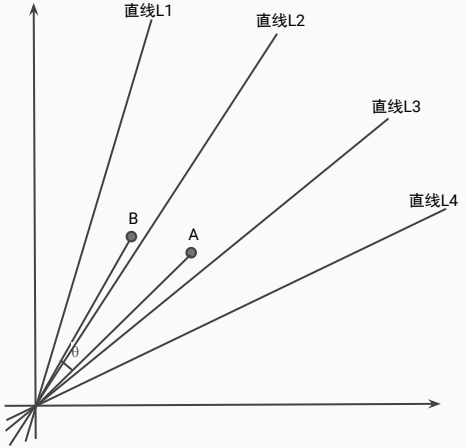
用坐标轴刻度坐标精确表示点的坐标



用相对于直线L1的位置近似表示点的坐标



直线越多，近似表示的结果越精确



随机创建一条过原点的直线，夹角为θ的两向量被分隔的概率为θ/π

当直线的数量N足够大时，可用 分隔的直线数数/N*π 来近似估算θ

如左图所示，压缩表示后：

A的坐标为(0, 0, 1, 1)

B的坐标为(0, 1, 1, 1)

四条直线中有一条将两向量分隔，所以 θ可近似表示 为0.25π

即压缩表示后A、B向量的汉明距离可近似表示 θ

汉明距离为两二进制数中不同的bit位数

使用LSH后，查询过程如下：

1. 对查询向量进行压缩编码
2. 计算库内的压缩编码与查询编码的汉明距离取最小的 topk

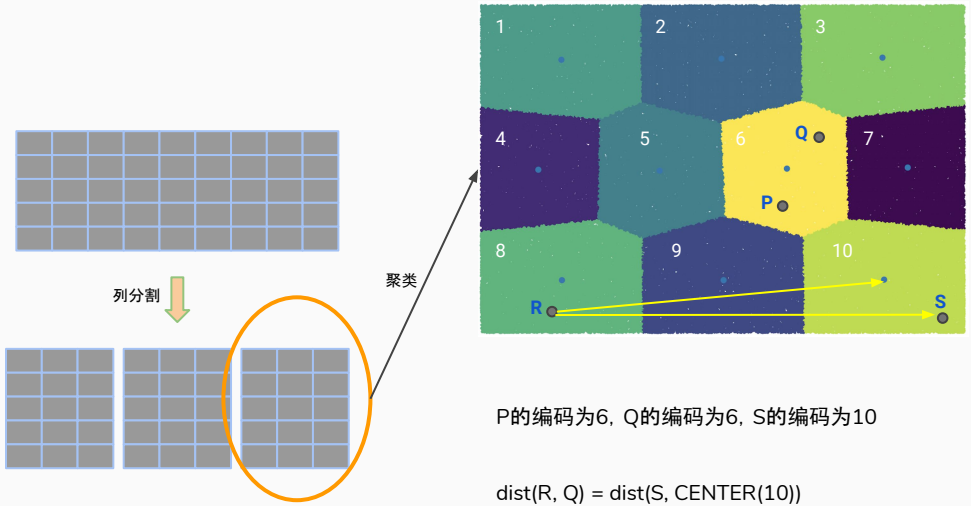
LSH的优点：

1. 由于汉明距离的计算非常快，所以就算是遍历数据库也不会很慢

LSH的缺点：

1. 因为需要扫库，不适用于超大规模的数据集。
2. LSH在实现的时候为了考虑执行效率，选择的划分粒度非常粗，不适用于高精度的场景

PQ(Product Quantizer)



P的编码为6，Q的编码为6，S的编码为10

dist(R, Q) = dist(S, CENTER(10))

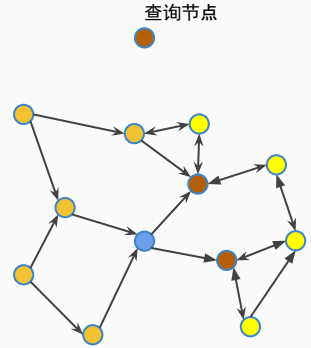
PQ的优点：

1. 计算距离时近似使用查询向量与聚类中心点的距离，查询速度快
2. 能对空间进行非常灵活的划分，精度 远高于LSH

PQ的缺点：

1. 因为需要扫库，不适用于超大 规模的数据集

NSW



图中的每个点都指向与其最近的2个点

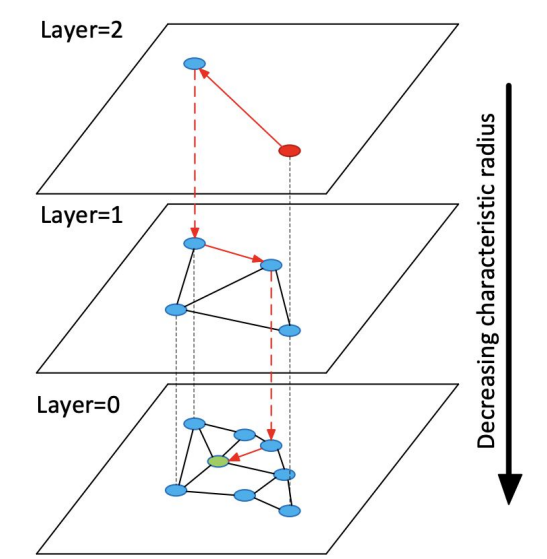
对于任一给定的连通图，且图中每一节点指向与其最近的K个节点，给定任一点，有如下查询方法：

1. 随机选择图中的一个节点作为初始节点
2. 将当前节点的K个最近节点放入备选列表
3. 计算与备选列表中每一节点的距离
4. 逐一获取备选列表中每一节点的最近K个节点，放入计算列表中
5. 计算查询节点与计算列表中每一节点的距离
6. 选取最近的K放入备选列表，截取备选列表的前K个
7. 如果备选列表发生变化，则停止查询并返回结果，否则继续执行第4步

图的构建过程：

1. 对所有待插入的节点，在图中查询最近的K个节点
2. 将插入的节点与最近的K个节点连接，完成一次插入

HNSW



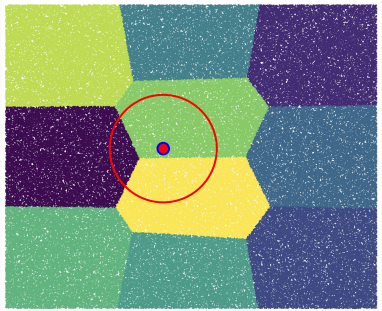
Layer = floor(-ln(random(0, 1)) * ml)

ml为可调整参数

优点：查询速度快、精度高

缺点：需要保存 邻居节点的指针，内存消耗大，不适用于大数据集和内存 紧缩的场景

IVF(Inverted File)



将数据分桶存储，查询时只在相邻的区域内召回候选集

数据集划分的方法可PQ、LSH等

谢谢！