DENSE RETRIEVAL WITH APACHE SOLR NEURAL SEARCH

Introduction

Lucene has introduced vector indices since 2020. Our work consists of exploring dense vector search using Apache Solr.

This paper presents the use of dense vectors for neural search and their integration in Apache Solr.

Overview

The main class introduced in the solr implementation is DenseVectorField. This class inherits from FloatPointField.

Index Field

The vector is stored using the parent class. The float values of the vector are stored as multivalue FloatPointField.

Query Parser

To search over the vector index, it is necessary to use the KnnQueryParser. It takes 5 parameters:

- Field name - the name of the field containing the vector.
- Vectors - a list of vectors to perform the search.
- TopK - the number of vectors to return.
- Indexing - whether the indexing is done in the Solr index.
- Query vector - a query vector to search against.

Field type parameters

- Similarity Function
  - EUCLIDEAN - default
  - DOT_PRODUCTS
  - COSINE

- Vector Dimension
- Indexing - whether the indexing is done in the Solr index.
- Query vector - a query vector to search against.

Benchmarks

- Test index size: 1.54 GB
- Vector index size: 1.17 GB

- # Documents: 4 46k
- # Queries: 575 0
- Avg Document Length: 108 7 Words
- Avg Query Length: 5.9 Words

Future Work

- Import latest Lucene changes into Solr.
- Support for filtering in KNN search.
- Up to 30% improvement in index throughput for vectors.
- Up to 10% faster KNN search.
- Dense vector fields Multivaled.
- Index/query time BERT integration in Apache Solr.

References


Reranking (to be improved)

K-Nearest Neighbor search

Fast indexing of Dense Vectors

Hierarchical Navigable Small Worlds (HNSW)

To search over the vector index, it is necessary to use the KnnQueryParser. It takes 5 parameters:

- Field name - the name of the field containing the vector.
- Vectors - a list of vectors to perform the search.
- TopK - the number of vectors to return.
- Indexing - whether the indexing is done in the Solr index.
- Query vector - a query vector to search against.

Field type parameters

- Similarity Function
  - EUCLIDEAN - default
  - DOT_PRODUCTS
  - COSINE

- Vector Dimension
- Indexing - whether the indexing is done in the Solr index.
- Query vector - a query vector to search against.

Benchmarks

- Test index size: 1.54 GB
- Vector index size: 1.17 GB

- # Documents: 4 46k
- # Queries: 575 0
- Avg Document Length: 108 7 Words
- Avg Query Length: 5.9 Words

Future Work

- Import latest Lucene changes into Solr.
- Support for filtering in KNN search.
- Up to 30% improvement in index throughput for vectors.
- Up to 10% faster KNN search.
- Dense vector fields Multivaled.
- Index/query time BERT integration in Apache Solr.

References


Reranking (to be improved)

K-Nearest Neighbor search

Fast indexing of Dense Vectors

Hierarchical Navigable Small Worlds (HNSW)

To search over the vector index, it is necessary to use the KnnQueryParser. It takes 5 parameters:

- Field name - the name of the field containing the vector.
- Vectors - a list of vectors to perform the search.
- TopK - the number of vectors to return.
- Indexing - whether the indexing is done in the Solr index.
- Query vector - a query vector to search against.

Field type parameters

- Similarity Function
  - EUCLIDEAN - default
  - DOT_PRODUCTS
  - COSINE

- Vector Dimension
- Indexing - whether the indexing is done in the Solr index.
- Query vector - a query vector to search against.

Benchmarks

- Test index size: 1.54 GB
- Vector index size: 1.17 GB

- # Documents: 4 46k
- # Queries: 575 0
- Avg Document Length: 108 7 Words
- Avg Query Length: 5.9 Words

Future Work

- Import latest Lucene changes into Solr.
- Support for filtering in KNN search.
- Up to 30% improvement in index throughput for vectors.
- Up to 10% faster KNN search.
- Dense vector fields Multivaled.
- Index/query time BERT integration in Apache Solr.

References


Reranking (to be improved)

K-Nearest Neighbor search

Fast indexing of Dense Vectors

Hierarchical Navigable Small Worlds (HNSW)