Methods of Retrieving the Most Accurate Context from Keywords

Amin Taheri

August 2, 2024

1 Introduction

This document surveys various methods to accurately retrieve text with specific contextual relevance to two topics: **Diversity** and **Data Security**.

2 Possible Methods

2.1 Traditional Keyword Matching

For the topic of Diversity, keywords such as ”diversity,” ”workplace,” ”inclusion,” etc., would be used to locate relevant text. Similarly, for Data Security, keywords like ”privacy,” ”cybersecurity,” ”compliance,” etc., would be employed.

Advantages:

- Easy to implement.
- Fast processing time.

Disadvantages:

- Low accuracy.
- Misses out on relevant text .

2.2 Semantic Search with Pre-trained Models

Using pre-trained models like BERT, RoBERTa, or Sentence Transformers to understand the context of the text rather than just keyword presence

Advantages:

- High accuracy.
- Finding relevant text from the context

Disadvantages:

- Computational resources.
- Needs fine-tuning sometimes

2.3 Word2Vec (w2v) Based Approach

W2V can be used to find semantically similar texts to the keywords of interest, even if they don’t share the exact words.

Advantages:

- Captures semantic similarity between words.

- Efficient for finding contextually related texts.

Disadvantages:

- Less effective compared to deep learning models like BERT.
- Requires a large corpus for training to achieve high-quality embeddings.

3 Recommended Approach

Among the methods discussed, **Semantic Search with Pre-trained Models** is recommended as the most accurate solution. Based on this link there are two methods on semantic search:

3.1 Symmetric vs. Asymmetric Semantic Search

A critical distinction for your setup is symmetric vs. asymmetric semantic search: For symmetric semantic search, your query and the entries in your corpus are of the same length and have the same amount of content. An example would be searching for similar questions: Your query could be "How to learn Python online?" and you want to find an entry like "How to learn Python on the web?". You could flip the query and the entries in your corpus for symmetric tasks.

- Related training example: Quora Duplicate Questions.
- Suitable models: Pre-Trained Sentence Embedding Models

For asymmetric semantic search, you usually have a short query **(like a question or some keywords)** and you want to find a longer paragraph answering the query. An example would be a query like "What is Python" and you want to find the paragraph "Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy . . . ". For asymmetric tasks, flipping the query and the entries in your corpus usually does not make sense.

- Related training example: MS MARCO
- Suitable models: Pre-Trained MS MARCO Models

Based on this link, 'msmarco-distilbert-base-v4' is better performing, so we are going to use it. It is using distillbert for its architecture which uses distillation techniques to make bert faster and smaller.