EMORY UNIVERSITY

Optimizing Search Efficiency: Exploring Differentiable Search Index Models for Information Retrieval

Information Retrieval

May 8, 2024

Navid Azimi, Alireza Rafiei

Department of Computer Science, Emory University

Contents

Introduction

Motivation

- The efficacy of IR systems in providing pertinent document rankings in response to user queries is paramount
- Traditional IR systems commonly employ an index-then-retrieve pipeline (not always the most efficient approach!)
- Alternative approaches like the Differentiable Search Index (DSI) aim to integrate indexing and retrieval processes into a unified model. By doing so, they offer the potential for more seamless and efficient document ranking in response to user queries.

Problems

- **Sequential nature of traditional IR methods:** Indexing occurs first, followed by retrieval
  → Latency issues, especially when dealing with large datasets or in real-time search scenarios
- **Struggle with handling dynamic or evolving datasets:** The index becomes outdated over time
  → Stale search results and diminish the user experience
- **Futile optimization efforts:** Improvements made in one stage may not directly benefit the other
  → Limit the system's ability to adapt and improve over time.

Goal

- Inspired by the Differentiable Search Index concept, our goal is to merge these traditionally separate stages and developing a unified model, designated as 'f', utilizing a sequence-to-sequence architecture, which handles user queries ('q') and employs an auto-regressive approach to generate relevant document IDs.

Contents

Dataset

The MS MARCO dataset → Pre-built index provided by **Pyserini** library

1 Building Resources

- **documents (dictionary):** document ID (docid) as key, a dictionary with field 'raw' (containing the raw text as string) as value
- **queries (dictionary):** query ID as key , a dictionary with field 'raw' (containing the raw text as string) and 'docids_list' (containing the list of correlated document IDs) as value

2 Computing Word2Vec Embeddings for queries and documents

- Using the trained Word2Vec model (on corpus data, containing all the processed 'raw' data
- Processed 'raw' data: Converted to lowercase, tokenized, and with stopwords and punctuation removed
- Created 2 types of embeddings for each query/document: emb, obtained as the average of the embeddings of all words, and first_L_emb, obtained by concatenating the embeddings of the first MAX_TOKENS words

3 Creating Datasets for Siamese Models

- **Pairwise Dataset (query, document, relevance):**
  → Designed for pairwise learning, where each sample consists of a query paired with a document
- **Triplet Dataset (query, doc+, doc-):**
  → Designed for triplet learning, where each sample comprises a query along with a positive and a negative document for training

Dataset

Example of Pairwise/Triplet Datasets

4 Creating Datasets for Sequence-to-Sequence (seq2seq) Models

- **Document Dataset (encoded document, encoded docid):**
  → Designed to facilitate training Seq2Seq models by providing documents as input sequences
- **Retrieval Dataset (encoded query, encoded docid):**
  → Designed for Seq2Seq models used in retrieval tasks. It pairs documents with corresponding queries for training

Documents and queries encodings are computed using the **T5-small tokenizer**, and we choose to pick the first **L=32** tokens for representing the documents, and the first **L=9** for representing the queries.

Example of Document/Retrieval Datasets

Contents

Baseline Models: Siamese Neural Networks

Convolutional Siamese Neural Network Baseline

Baseline Models: Siamese Neural Networks

Siamese Transformer Baseline

Baseline Models: Siamese Neural Networks

Siamese Triplet Baseline

Contents

DSI Transformer-Based Model

DSI Transformer-Based Model

DSI Transformer-Based Model

Contents

Conclusion

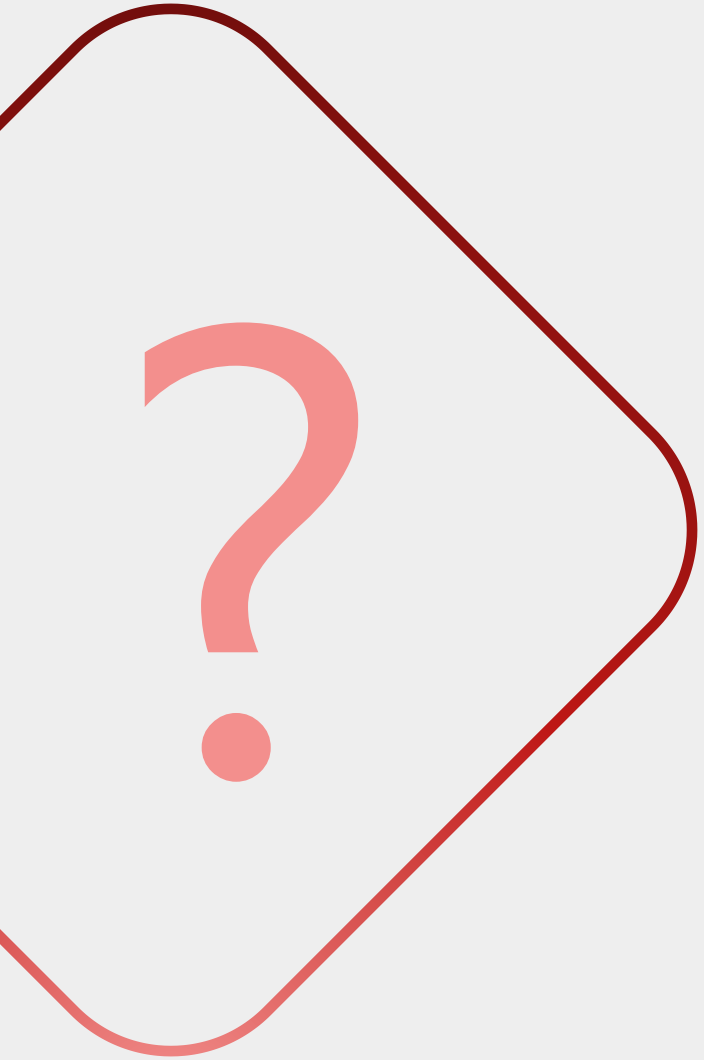
References

1 **Transformer memory as a differentiable search index**

Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, Tal Schuster, William W. Cohen, and Donald Metzler.

2

3



Thank You!