Documentation Search with Elasticsearch and GENRE

Team

- Ilya Andreev (captain), iandre3@illinois.edu
- Raj Krishnan, rajk3@illinois.edu
- Sengill Kim, sk100@illinois.edu

Abstract

In this project, we will compare Generative Entity Retrieval and standard full text search methods for looking up knowledge base entities. We use Python documentation as the knowledge base, documentation sections (e.g. “Instance Objects” or “Expression Lists”) as documents, and StackOverflow-like questions as queries. We answer these queries in two ways: by retrieving relevant documentation with Elasticsearch and by generating relevant documentation section names.

Proposal

A paper on Autoregressive Entity Retrieval came out in March 2021, along with the accompanying implementation called GENRE. This paper offers an alternative approach to searching knowledge bases. It uses a sequence-to-sequence model to handle end-to-end entity linking, entity disambiguation and document retrieval. In practice, this means that GENRE generates names of the knowledge base entries related to the query based on the context in which those entries have been referenced in the training data.

We seek to compare this approach with an industry standard search solution (Elasticsearch, which is built on Apache Lucene) and see how the two approaches differ for the task of retrieving entities from documentation when presented with StackOverflow questions as queries. We find this work important since it compares some of the classical approaches that we have covered in the class with more novel but less common methods.

For this project, we are going to build our own training and test datasets. These datasets will consist of publicly available StackOverflow questions with popular answers linking to specific language documentation sections. We are going to use Python 3 Standard Library and Language Reference as our knowledge base. All systems implemented for this project will be written in Python.

Our implementation steps will roughly involve the following:

1. Build a corpus of language documentation.
2. Index the language documentation into Elasticsearch with minimal tuning.
3. Build a corpus of queries from StackOverflow, retrieving user questions with at least one popular accepted answer. Filter the results to only consider the answers linking to Python documentation, and split the question-answer pairs into a training and test dataset.
4. Use the training dataset from StackOverflow to train the GENRE sequence-to-sequence model.
5. Run the test queries against the model based on GENRE and Elasticsearch, and see how their Mean Reciprocal Rank metrics compare.

We expect the workload to be distributed as follows:

1. Build a language documentation corpus: 5 hours
2. Index documentation into Elasticsearch: 5 hours
3. Build StackOverflow question-answer dataset: 20 hours
4. Train the sequence-to-sequence model using the StackOverflow dataset: 20 hours
5. Build a test harness to evaluate approaches: 15 hours
6. Compare approaches: 2 hours
7. Total: 67 hours