TensorFlow Tools for Text Retrieval Problems

Technology Review, CS410, Fall 2021

Sofia Godovykh

Abstract

TensorFlow is one of the most advanced Machine Learning tool. TensorFlow is mostly focused on Deep Learning, but its functions and even pre-trained models can be quite helpful for traditional NLP. In this article I will describe TensorFlow features for NLP and text retrieval. I will cover text processing tools; ML options for Natural language processing; and, finally, existing datasets and models.

Text processing

Text processing is the first step in solving machine learning problems. In many cases software engineers and researchers deal with raw data, which needs to be cleaned, processed and transformed in order to provide further NLP steps the best data possible. While it can be done with traditional algorithms like Porter or Lancaster Stemmer, TensorFlow provides its own implementation. Here is the list of some available modules: **TextVectorization** handles classic NLP tasks, such as lowercase, remove punctuation, build ngrams and tf-idf. **WhitespaceTokenizer**, the most basic string tokenizer which splits string into individual words. **WordpieceTokenizer** is more complex and works as stemmer. The tokenizer uses dictionary, built with top-down WordPiece generation algorithm. There are four steps in building such dictionary: 1. Count word frequency (w, c) for the provided document. 2. For each given word generate a set of all possible substrings: *word -> ["w", "wo", "wor", "word", "#ord", "##rd", "###d"]*. 3. Build a hashmap *(S, TC)* for substring, where *TCk = sum(C0, C1, ..., Cn)*, where *Ci* is a count of word *Wi*, which generated substring *k*. 4. Remove substrings *Si* which have *TCk* less than given threshold *T*. 5. For each of the substring left, subtract off its count from all of its prefixes. **BertTokenizer** combines the previous two, but also performs additional tasks such as normalization. **SentencepieceTokenizer** is a sub-token tokenizer that is highly configurable. This is backed by the Sentencepiece

library. Like the BertTokenizer, it can include normalization and token splitting before splitting into sub-tokens. TensorFlow provides ready-to-use vocabularies, but it is also possible to create a custom one . Out-of-vocabulary word is a common issue for tokenization, so pre-trained tokenizers are usually perform better.

Natural Language Processing tools

Here I cover two major functions used in text retrieval: text classification and ranking. TensorFlow has comprehensive, highly customizable tools for both problems.

1. Ranking

In TensorFlow, ranking is a part of TensorFlow Recommenders (TFRS) module. It helps with the full workflow of building a recommender system: data preparation, model formulation, training, evaluation, and deployment. Module has a variety of pointwise, pairwise, and listwise loss functions. It provides Mean Reciprocal Rank, Discounted Cumulative Gain, Normalized Discounted Cumulative Gain (NDCG), Mean Average Precision and other popular metrics. It also has LambdaLoss, a probabilistic framework for ranking metric optimization. Such metric-driven loss functions significantly improve ranking algorithms. Another component is Unbiased Learning-to-Rank from biased feedback data, which implements ranking algorithm itself. In May 2021 Google released a major update of TFRS. It uncluded ModelBuilder, DatasetBuilder, and a Pipeline, tools for set up and train the model with the provided dataset. The key feature of the update was new TFR-BERT architecture, a combination of learning-to-rank (LTR) models and Bidirectional Encoder Representations from Transformers (BERT) representation. Google also introduced neural ranking generalized additive models (GAM) for ranking problems requiring interpretability.

2. Classification

Classification is not the best method for solving text retrieval problems, but it can be used for narrowing down the selection of documents. Recurrent neural network (RNN) is an advanced classification algorithm which can not be found in other NLP and ML libraries such as NLTK and Scikit-learn. RNN is a neural network that is intentionally run multiple times, where parts of each run feed into the next run. Specifically, hidden layers from the previous run provide part of the input to the same hidden layer in the next run. Recurrent neural networks are particularly useful for evaluating sequential text data, so that the hidden layers can learn from previous runs of the neural network on earlier parts of the sequence.

Existing dataset and models

1. Datasets

There are over a hundred datasets with TensorFlow, I have selected ones relevant for text retrieval.

- **ag_news_subset** is a collection of news articles labeled by topic.
- **eraser_multi_rc** is a datest, each datapoint of which has a text; a query about it; answer; whether the answer is right or wrong; and an explanation justifying the classification
- **qa4mre c**ontains a passage, a set of questions corresponding to the passage, and multiple options for answers are provided for each question, of which only one is correct.
- **trec** is a question classification dataset with 5500 labeled questions.
- **istella** is a set of query-document pairs represented as feature vectors and corresponding relevance judgment labels.
- **mslr_web** have the same structure as istella, but at smaller scale.

2. Models

- **DLRM and DCN v2** are both ranking models. The model inputs are numerical and categorical features, and output is a scalar.
- **Neural Collaborative Filtering** (NCF) is a general framework for collaborative filtering of recommendations in which a neural network architecture is used to model user-item interactions.

Conclusion

TensorFlow has tools for each step and aspect of NLP problems. It is a powerful and highly customizable framework, supported by Google and community of machine learning researchers and enthusiasts. TensorFlow ecosystem includes comprehensive documentation, visualisation and debugging tools.

References

https://www.tensorflow.org
https://github.com/tensorflow
https://ai.googleblog.com/2021/07/advances-in-tf-ranking.html