Ranking with Neural Network Derived Document Vectors

Anton Serjogin
Centre for Information and Speech Processing, LMU
anton.serjogin@gmail.com

15.05.2017

Motivation: help students find the best suited course for them by using a search engine.The work is based on developing an intelligent MOOCS search engine, where neural networks are used. The IROM project of the institute of informatics is the framework where this search engine is being developed and tested.

The principle of this search engine is: a certain user has a need for information, he types in the word query, after which a specific algorithm goes through domains and gives out the result as ranked recommendations. An interesting feature is that this algorithm takes the user's metadata into account, for instance, click-through data. As this search engine tries to deliver high-accuracy results, several optimizations are used for the query, as well as better ranked recommendations are achieved by using Deep Relevance Matching models.

The axiom is efficient document representation, which should be both effective and efficient. A traditional model in Information Retrieval is TF-IDF - a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus, which is often used as a weighting factor. There are, however, two drawbacks: flawed word independence assumption and ignorance of word order. Therefore, an artificial neural network, known as autoencoder, is used, which is good for unsupervised learning, it has also the ability to predict next words. In order to achieve a semantic space for documents, Word2Vec and Doc2Vec are used as well. One the traditional approaches features LSA (Latent Semantic Analysis), however, the problem is that it should be done on an entire corpus, making it useful only offline.

Recurrent networks can learn to compress whole history in low dimensional space, while feedforward networks compress just single word, another advantage is that a recurrent network's history is represented by neurons, making the history length unlimited. Documents are created with the RNN and Seq2Seq is used to extract feature vectors from documents. An example of the high dimensional vector space including the documents as feature vectors was shown in Plotty.

The whole work includes such steps as:

- Train LSTM Seq2Seq
- Create API
- Evaluate ranking performances
- Evaluate selected features
- Search for constant shifts (bonus)