TEXT ANALYSIS BASED ON LSTM MODEL WITH PRE-TRAINED WORD2VEC EMBEDDING

Kanishchev I.S.

Vyatka State University,
Russia, 610000, Kirov, Moskovskaya st., 36,
E-mail: kanishchev.ilya@gmail.com

While the role of genetic testing in advancing our understanding of cancer and designing more precise and effective treatments holds much promise, progress has been slow due to significant amount of manual work still required to understand genomics. For the past several years, world-class researchers at Memorial Sloan Kettering Cancer Center have worked to create an expert-annotated precision oncology knowledge base. It contains several thousand annotations of which genes are clinically actionable and which are not based on clinical literature.

There are nine different classes into which a genetic mutation can be classified. There are 3321 different identifiers in the training set, containing 264 different gene expressions and 2996 different mutation variations. The test sample contains 70% more data. A big part of the problem is to teach an ML model how to "read" medical literature and classify the given Gene and Variation into 1 out of 9 classes.

The training model is based on the LSTM with pre-trained word2vec embedding in Keras. Word2Vec is one of the most popular technique to learn word embedding using shallow neural network. It was developed by Tomas Mikolov in 2013 [1]. An embedding is a mapping from discrete objects, such as words, to vectors of real numbers. Embedding are important for input to machine learning. Classifiers, and neural networks more generally, work on vectors of real numbers. They train best on dense vectors, where all values contribute to define an object. However, many important inputs to machine learning, such as words of text, do not have a natural vector representation. Embedding functions are the standard and effective way to transform such discrete input objects into useful continuous vectors.

The results of the quick LSTM are promising. On the first try with no hyper parameter search, 6th epoch, max sequence length cut down to a measly 2000 (longest text has 77000+ words), we get the best log loss so far of around 1.4. The results are still not very good though. One way to explain this is that there is a lot of information loss from just getting the mean of all word vectors of the document. Further tuning of the LSTM will likely produce better results.

References

1. *Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean* Distributed Representations of Words and Phrases and their Compositionality. - Advances in Neural Information Processing Systems 26, 2013
2. *Andrew Trask, David Gilmore, Matthew Russell* Modeling Order in Neural Word Embeddings at Scale. - ICLR, 2016