Introduction

This paper is about distributed representation embedding for words, as an extension to the skip-gram model. Since our project involves implementing a deep learning model for fake news detection, I am interested in finding models to create word embedding to represent words efficiently.

Model Analysis

This model allows capturing of millions of phrases, instead of individual words so this way relationships are represented. Representing phrases as vectors allows for a degree of compositionality that is not trivial (ie adding/subtracting phrases for meaningful results). This appears very useful for predictions and our project since we can have access to a word embedding layer that is efficient and has meaning beyond just each individual word token.

The skip-gram model is enhanced with various features/options such as hierarchical softmax and subsampling and negative sampling.

Hierarchical softmax: Skip-gram model attempts to maximize the probability of a word and its surrounding context. One improvement is to use a binary tree representation of the output layer with W words as the leaves, and for each node, represent the relative probability of its child.

Subsampling: Subsampling of frequent words allows for the model to learn word pairs where frequent co-occurrences are thrown out. This allows the model to learn word pairs that are non-trivial, ie toss out words like “the” in pair “France the” and keep “France Paris”. This feature of the skip-gram model is very interesting and reminds me (although is different) of document inverse frequency as taught in the lectures. Less frequent words add more value!

Negative sampling: the training sample only modifies a small random portion of the weights. A random portion of the negative samples/ words are changed .The positive word weight is changed, but the rest of the negative samples remain unchanged. This way, the training time is greatly improved.

Model Empirical Results

The final skip-gram models and its various features were trained on 33 billion words. Hierarchical softmax with dimensionality of 1000 combined with subsampling to result in the best accuracy of 72%. This is very respectable since many meaningful phrases were learned. The amount of words the authors used to train this model also seems huge! And bigger than any of the other models that I heard of .

The phrase representations learned by skip-gram model allow for vector compositionality, since words were trained to predict surrounding words. Thus words can be combined by adding vectors for country + currency combinations, country + capital and etc. I think this is a very

feature of the skip-gram model, since other bi-gram models can do not have this linear feature, and can not meaningfully combine words.

Concluding Thoughts

The skip-gram model seems to have very high- quality phrase representations; It can learn very unique word combinations and phrases. Compared to uni-gram models, the phrases capture more meaning/context which is very difficult and impressive. Word vectors can also be meaningfully combined to produce reasonable words/phrases based on their context.

This model seems very unique in that it considers the context of surrounding words and learns meaningful phrases. I am definitely interesting in using this model in the future! Perhaps we may use a pre-made embedding layer for sentiment analysis work.

Some of the original issues with skip-gram were computation time and size due to the number of neurons and weights for each word. This made it very slow to train to model.

However, the authors provide enhancements such as hierarchical softmax, negative sampling that actually greatly improved the training time, and made this method much faster than others.