Paper: *Word2Vec Tutorial - The Skip-Gram Model*

Razwan Ahmed Tanvir

Date: 01/20/2022

Quote *Using a word offset technique where simple algebraic operations are performed on the word vectors, it was shown for example that vector("King") - vector("Man") + vector("Woman") results in a vector that is closest to the vector representation of the word Queen.* pp.2.

Overview The author of the blog took an attempt to discuss the skip gram neural network architecture for Word2Vec. The model described in the blog, uses a simple neural network with a single hidden layer which is used to learn the weights of the hidden layer. The goal of the learned weights is to output a set of nearby words relating to the input word. This Word2Vec model transforms word to its vector representation and also reserves the similarity between related words. It is done by back propagation of the generated output and the expected output of the model. This model solves one of the basic properties of natural language which is the context of the word in a sentence. Word2Vec model solves this problem by generating similar vectors for similar words. For example, the word "cat" and "dog" will have small cosine distance of their vectors.

One of the major accomplishments of this model is that, the paper [1] introducing the model, works with very large corpus. This large amount of data ensures high quality word vectors. Another notable accomplishment of this Word2Vec model is that, we can establish algebraic relation between similar words. For example, the vector("King") - vector("Man") + vector("Woman") results in a vector that is closest to the vector representation of the word "Queen".

Intellectual Merit Previously proposed models were trained on few hundered of millions of word, but the model proposed in the referenced paper, trained their model using 1.6B word which is huge size of a dataset and it can generate high quality word vectors. This research advanced the understanding of the similarities of the word vectors which can help in machine translation, speech recognition and so on. These creative concepts boost the progress of these mentioned tasks and improve the state-of-the-art architectures. The accuracies of this research were compared to the state-of-the-art results which ensures the reasoning of the results of this research. This Word2Vec model was developed in Google and Google is undoubtedly qualified for these types of research for their access to the large amount to text data used to train the Word2Vec model. The authors of the referenced paper have access to the Google's data and also the state-of-the-art hardware to train the model. The model can be trained under one day, the authors mentioned in the paper.

Broader Impact This research has several impacts on the upbringing of text data analysis. Multiple degrees of similarity, as mentioned in the paper, has impact on several natural language processing based applications like machine translation, text-to-speech etc. This publication is publicly available and open to explore. Moreover, the datasets and code are available for use. However, all the authors of this paper are employees of Google and they are all male.

Keywords Natural Language Processing, Neural Network, Skip-Gram model

Discussion Questions

- The blog discusses about the weight matrix. However, it was not shown that how to actually calculate the weight matrix by back propagation. The weight matrix is the sole decider of a word vector in Word2Vec model. The blog says that the hidden layer will learn the weights of the layer by back propagation but, there's no mention of how the model actually calculates the weight matrix.

- This blog explains the Wrod2Vec model very good, however, it does not include any coding example or code snippets. Therefore, though this blog explains the model, but the reader cannot actually feel how the model would be implemented in real life.

- The blog says, that there is a window, but it is not clear to me whether all the generated words for an input word have the same probability?

References

[1] Mikolov, Tomas & Chen, Kai & Corrado, G.s & Dean, Jeffrey. (2013). Efficient Estimation of Word Representations in Vector Space. Proceedings of Workshop at ICLR. 2013.

Table 1: Grade deductions by section

Overview	Intellectual M.	B. Impact	Keywords	Questions	Is Online?