Word2Vec For Word Embeddings

# Why Word Embeddings?

Let consider the 2 sentences --- “You can scale your business” and “You can grow your business”. These 2 sentences have the same meaning. If we consider a vocabulary considering these 2 sentences, it will continue of these words: {You, can, scale, grow, your, business}

A one-hot encoding of these words would create a vector of length 6. The encodings for each of the words would look like this:

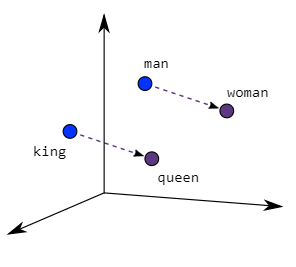
You: [1,0,0,0,0,0], Can:[0,1,0,0,0,0], Scale:[0,0,1,0,0,0], Grow:[0,0,0,1,0,0], Your:[0,0,0,0,1,0], Business: [0,0,0,0,0,1]

In a 6-dimensional space, each word would occupy one of the dimensions, meaning that none of these words has any similarity with each other --- irrespective of their literal meanings.

Word2Vec, a word embedding methodology, solves this issue and enables similar words to have similar dimensions and, consequently, helps bring context*.*

# What is Word2Vec?

Word2Vec creates vectors of the words that are distributed numerical representations of word features --- these word features could comprise of words that represent the context of the individual words present in our vocabulary. Word embeddings eventually help in establishing the association of a word with another similar meaning word through the created vectors.

As seen in the image below where word embeddings are plotted, similar meaning words are close in space, indicating their semantic similarity.

Two different model architectures that can be used by Word2Vec to create the word embeddings are the Continuous Bag of Words (CBOW) model & the Skip-Gram model.

# How does CBOW work?

Even though Word2Vec is an unsupervised model where you can give a corpus without any label information and the model can create dense word embeddings, Word2Vec internally leverages a supervised classification model to get these embeddings from the corpus.

The CBOW architecture comprises a deep learning classification model in which we take in context words as input X, and try to predict our target word Y.

A diagram of a algorithm

Description automatically generatedFor example, if we consider the sentence --- “***Word2Vec has a deep learning model working in the backend***”, there can be pairs of context words and target (center) words. If we consider a context window size of 2, we will have pairs like ([deep, model], learning), ([model, in], working), ([a, learning], deep) etc. The deep learning model would try to predict these target words based on the context words.