Bag Of Words

Rodrigo Rocha Amorim

June 2020

1 Introduction

Natural Language Processing it's a field that process many kind of natural language, like speech and text, by software. In the field of machine/deep learning, NLP it's vastly used for task like sentimental analysis, modelling language, machine translation, speech recognition and others.

For using models of machine and deep learning it's best recommended if is used **vectors of numbers to feed in this algorithms, so there is a need to convert in the case of text, raw text, into this number vectors**. The process of convert the text into number vectors is called feature extraction or feature encoding. A popular example is Bag of words;

In This brief tutorial will be explain what is Bag of Words, why it's used, how is build and your limitations.

For practical content, follow me on github: RodrigoAmorimml

2 Bag of Words

One problem with modelling text is that text is messy, and techniques like **machine learning and deep learning algorithms prefer well defined fixed length inputs and outputs.** Meaning that the raw text needs to be converted into vectors of numbers, where this process is called feature extraction, and one way of doing this is a called Bag of Words.

A Bag of Words its a way of extracting features from texts, **it's a representation of text that describes the occurrence of word within a document.** It involves two steps:

2.1 A vocabulary of known words

To build a bag of words its need to create a vocabulary, the vocabulary has the size of number of words in the corpus, usually ignoring punctuation. Then its create a document vector to score the words in the document. The objective is to transform every sample (document, text, line) into a vector that can be used as input or output for a machine learning model. Usually it's used a fixed length vector, with one position in the vector to score each word, this score can

be done using the presence of words as a boolean value, 1 for presence and 0 for not present, then converting in a binary vector.

One problem that can be found is for huge documents, will be a huge vocabulary vector size, with a lot of zeros. This type of vector is called sparse vector, where it is a big vector with a lot of zeros and a fell ones. This vectors cost a lot of computation process and the vast number of positions or dimensions can make the modelling process more complex. One way for decrease its filtering punctuation, frequent words (stop words) and reducing some words to their stem.

Another possibility its create a vocabulary group of words, also called n-gram where n can be any number and gram represent the token word, for example, 2-gram or bigram its a vocabulary grouped by 2 words.

Importante things to know:

- **New documents that contains words that isn't in the vocabulary are ignored.**
- **Only the bigrams that appears in the corpus are modeled, not all possible bigrams.**
- **Usually a simple bigram approach is better than a 1-gram BoW for task like classification.**

2.2 A measure of the presence of known words

One way to score the words is counting the number of times each word appears in the document, the problem with this approach is that highly frequency words start to dominate in the document and this words may not contain much information for the model that will be used. Another way to measure is rescale the frequency of words to high frequencies words be penalized, this approach is called Term Frequency Inverse Document Frequency (TF-IDF). TF-IDF it's a way of weighting the words to equally all words.

3 Limitations of BoW

The limitations of bag of words, as was saw, is the vocabulary size, where it's need more data preparation for shrunk the size to decrease the impact of computational cost. Another problem is that BoW don't catch any meaning of the words, only score how many times appears in the vocabulary. This problem can be reduced using a gram of words bigger then 1, like a 2-gram.

4 Conclusion

This document explain the importance of using a text extraction feature for modelling machine and deep learning model, also the meaning of bag of words, how to build and their limitations.

5 References

[1.] Deep Learning for Natural Language Processing, Jason Brownlee, 2017.