**What is Bag of Words?**

It is a way of extracting features from the text for use in machine learning algorithms. We need a way to represent text data for machine learning algorithm and the bag-of-words model helps us to achieve that task. The bag-of-words model is simple to understand and implement. Here we use the tokenized words for each observation and find out the frequency of each token.

Let’s take an example to understand BOW:

“It was the best of times”“It was the worst of times”“It was the age of wisdom”“It was the age of foolishness”

Here we treat each sentence as separate document and we make a list of all words from all documents excluding the punctuation.

‘It’, ‘was’, ‘the’, ‘best’, ‘of’, ‘times’, ‘worst’, ‘age’, ‘wisdom’, ‘foolishness’

The next step is to create vectors. Vectors convert text that can be used by machine learning algorithm.

We take the first document — “It was the best of times” and we check the frequency of words from the 10 unique words.“it” = 1  
“was” = 1  
“the” = 1  
“best” = 1  
“of” = 1  
“times” = 1  
“worst” = 0  
“age” = 0  
“wisdom” = 0  
“foolishness” = 0

*Rest of the documents will be:*“It was the best of times” = [1, 1, 1, 1, 1, 1, 0, 0, 0, 0]“It was the worst of times” = [1, 1, 1, 0, 1, 1, 1, 0, 0, 0]“It was the age of wisdom” = [1, 1, 1, 0, 1, 0, 0, 1, 1, 0]“It was the age of foolishness” = [1, 1, 1, 0, 1, 0, 0, 1, 0, 1]

Here in this approach each word or token is called “gram”. Creating a vocabulary of two-word pairs is called a bigram model.

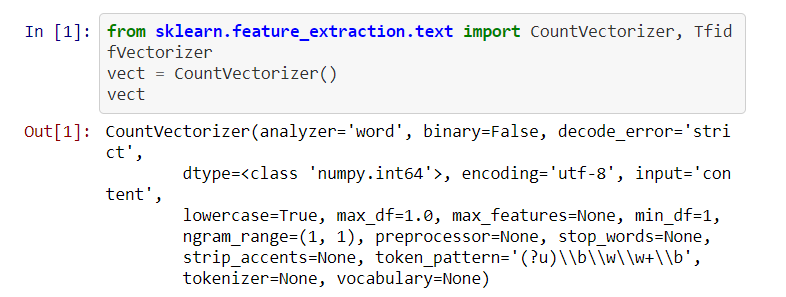
For example, the bigrams in the first document: “It was the best of times” are as follows:  
“it was”  
“was the”  
“the best”  
“best of”  
“of times”

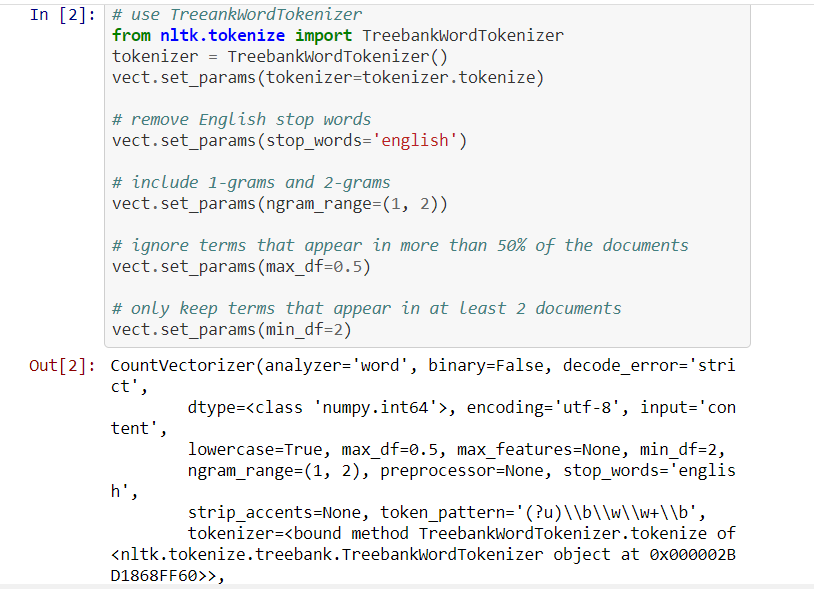
The process of converting NLP text into numbers is called **vectorization** in ML. Different ways to convert text into vectors are:

* Counting the number of times each word appears in a document.
* Calculating the frequency that each word appears in a document out of all the words in the document.

**Count Vectorizer**

It works on term frequency that is counting the occurrences of tokens and building a sparse matrix of documents x tokens.





**Drawbacks of BOW**

1. If the new sentences contain new words, then our vocabulary size would increase and thereby, the length of the vectors would increase too.
2. Additionally, the vectors would also contain many 0s, thereby resulting in a sparse matrix (which is what we would like to avoid)
3. We are retaining no information on the grammar of the sentences nor on the ordering of the words in the text.

**TFIDF**

TF-IDF stands for term frequency-inverse document frequency. TF-IDF weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus.

* **Term Frequency (TF)**: is a scoring of the frequency of the word in the current document. Since every document is different in length, it is possible that a term would appear much more times in long documents than shorter ones. The term frequency is often divided by the document length to normalize.

Image for post

* **Inverse Document Frequency (IDF)**: is a scoring of how rare the word is across documents. IDF is a measure of how rare a term is. Rarer the term, more is the IDF score.

Image for post

**Thus,**

Image for post

