1 Bag Of Words

1. Bag of words uses the idea that frequency of usage of words in one kind of documents (say postive product review) differs from frequency of usage of words in other kind of documents (say negative reviews). This is best exemplified by the sample image below:

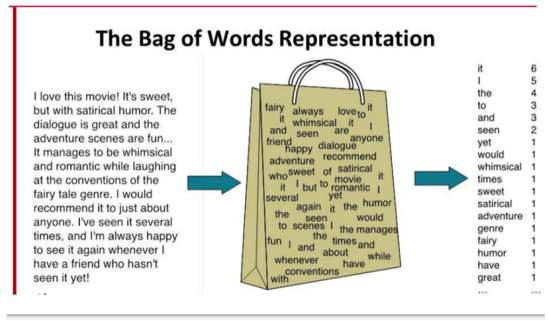

Figure 1

2. The training data, i.e. the vocabulary size, needs to be vast and diverse, to avoid the overfitting problem. This effort is only to minimize the risk of over-fitting, while its impossible to eliminate it completely.

3. Requires manual experimenting when deciding on the parameters that help form the valid words to be considered as the word-vector.The parameters such as:

   a) Most/Least frequent words: Most-frequent-words and least-frequent-words that needs to be eliminated from the bag of words.

   b) Frequency threshold: This needs to be tuned appropriately, so that we get accurate model, while avoiding over-fitting problem.

   Sample code below showing how valid words are derived after filtering out the most/least frequent words and by using the frequency threshold: (Note this is just a code snippet to provide a basic idea, and is not meant to be a complete implementation of Bag of words)

```
# Ignore the 25 most frequent words, and the words which appear less than 100 times
ignore_most_frequent = 25
freq_thresh = 100
feature_number = 0
for word, word_frequency in sorted_words[ignore_most_frequent:]:
    if word_frequency > freq_thresh:
        valid_words[word] = feature_number
        feature_number += 1
```

Figure 2

4. Since BOW piggy backs finally on the kNN algorithm for predictions, it is prone to the implmentation issues of kNN algorithm.