CSE 454 Data Mining
Final Project Report

Ahmed Semih Özmekik
171044039

February 1, 2021

Abstract

Detailed explanation of design choices along with the experimental results in the homework.

1 Preprocessing

First of all, as stated in the homework file, we combined the text classification data sets we have and the Turkish texts we used in the previous homework into a single file. In both texts, punctuation marks, some unnecessary characters, numbers, etc. there were many unwanted, noisy words. All of this is extracted from each dataset in the preprocessing phase.

word2vec was written to a single file, all cleared, in the format expected by its program. The preparation of this dataset was done just once, and we obtained another processed dataset from the raw datasets we have. As a result, we have large data text documents of more than 300MB in size. We save this every time in order not to process it from raw data sets. And now we're ready to extract the **word2vec**, vector, from this file.

2 Generating word2vec

We are performing the word2vec generation process with the simple code shown below. This stage takes about 15 minutes.

```
time ./word2vec -train ../train.txt -output vectors_400.bin
-min-count 0 -size 400
```

3 P calculations: 3 methods

We are asked to develop another strategy instead of counting classes, as in the classic Naive Bayes. Therefore, instead of a probability calculation consisting of class counts, we will make a calculation using the word2vec vector we have.

We selected 3 different simple methods for this and will be shared with their results.

3.1 Min Max Vector

In this section, we first find the **word2vec** of all the words in a document (ie a comment on the movie), then we get the maximum vector and the minimum vector of these vectors based on the column.

So let's be the length of a vector d. And let each word be a vector of v, v_w. The elements of the vector are as follows:

v= $[v_0, v_1, v_2, ..., v_{d-1}]$

Min (or max) produces a vector by taking the min of each column of this vector. The feature vector f we obtained for a document is as follows:

$f = [min(v_w) + max(v_w)]$

f has length of $2d$.

In short, we obtain a vector by merge the min and max values we get from **word2vec**.

3.2 Mean Vector

This method, just like the above, is a method of creating a feature vector for the document by taking the average of the vectors in the document.

A vector of length d is created.

3.3 TFIDF Vector

In this part, we first generate a TFIDF vector for all words. Then we multiply the word2vec vector found for each word with the TFIDF vector found for that same word to obtain the feature vector for a document.

With this process, we obtain a feature vector containing a mixture of TFIDF and word2vec.

4 Conclusion

The scores we have obtained can be viewed on the jupyter notebook we send in submission.

Method	F1-Score
Min-Max	0.6656
Mean	0.6731
TFIDF	0.7166