Lab Sheet 7 – Text Classification

Kavishka Gamage – 17000475

Screenshot of the result of Classifier evaluation demo.py

```
Actual counts: [('spam', 10), ('ham', 10)]
Predicted counts: [('spam', 11), ('ham', 9)]
             Predicted:
                    spam ham
Actual: spam           5   5
        ham            6   4
Accuracy: 0.45
Manually computed accuracy: 0.45
Precision: 0.45
Manually computed precision: 0.45
Recall: 0.5
Manually computed recall: 0.5
F1 score: 0.48
Manually computed F1 score: 0.47

(base) G:\Github\NLP\UCSC>
```

Screenshot of the result of feature extraction demo.py

```
(base) G:\Github\NLP\UCSC>C:/Users/Kavishka/anaconda3/python.exe "g:/Github/NLP/UCSC/Labsheets/2_5_Text Classification/feature_extraction_demo.py"
[[0 0 1 0 1 0 1 0 1]
 [1 1 1 0 2 0 2 0 0]
 [0 1 1 0 1 0 1 1 1]
 [0 0 1 1 0 1 0 0 0]]
[[0 0 1 0 0 0 1 0 0]]
['and', 'beautiful', 'blue', 'cheese', 'is', 'love', 'sky', 'so', 'the']
   and  beautiful  blue  cheese  is  love  sky  so  the
0    0          0     1       0   1     0    1   0    1
1    1          1     1       0   2     0    2   0    0
2    0          1     1       0   1     0    1   1    1
3    0          0     1       1   0     1    0   0    0
   and  beautiful  blue  cheese  is  love  sky  so  the
0    0          0     1       0   0     0    1   0    0
    and  beautiful  blue  cheese    is  love   sky    so   the
0  0.00       0.00  0.40    0.00  0.49  0.00  0.49  0.00  0.60
1  0.44       0.35  0.23    0.00  0.56  0.00  0.56  0.00  0.00
2  0.00       0.43  0.29    0.00  0.35  0.00  0.35  0.55  0.43
3  0.00       0.00  0.35    0.66  0.00  0.66  0.00  0.00  0.00
   and  beautiful  blue  cheese   is  love   sky   so  the
0  0.0        0.0  0.63     0.0  0.0   0.0  0.77  0.0  0.0
   and  beautiful  blue  cheese   is  love  sky   so  the
0  0.0        0.0   1.0     0.0  1.0   0.0  1.0  0.0  1.0
1  1.0        1.0   1.0     0.0  2.0   0.0  2.0  0.0  0.0
2  0.0        1.0   1.0     0.0  1.0   0.0  1.0  1.0  1.0
```

Screenshot of the result of Classification.py

Classes in news categories

```
['alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc', 'comp.sys.ibm.pc.hardware', 'comp.sys.mac.hardware', 'comp.windows.x', 'misc.forsale', 'rec.autos', 'rec.motorcycles', 're
c.sport.baseball', 'rec.sport.hockey', 'sci.crypt', 'sci.electronics', 'sci.med', 'sci.space', 'soc.religion.christian', 'talk.politics.guns', 'talk.politics.mideast', 'talk.politics
.misc', 'talk.religion.misc']
```

Result of SVM classifier and bow feature set

```
svm_bow  =>

Accuracy: 0.64
Precision: 0.67
Recall: 0.64
F1 Score: 0.65
```

Result of SVM classifier and TFIDF feature set

```
svm_tdidf  =>

Accuracy: 0.77
Precision: 0.77
Recall: 0.77
F1 Score: 0.76
```

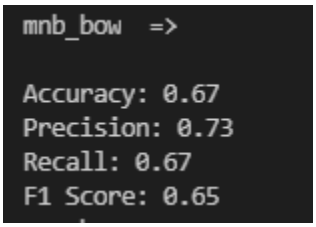
Result of SVM classifier and avg word2vec feature set

```
svm_avgwv  =>

Accuracy: 0.56
Precision: 0.57
Recall: 0.56
F1 Score: 0.55
```

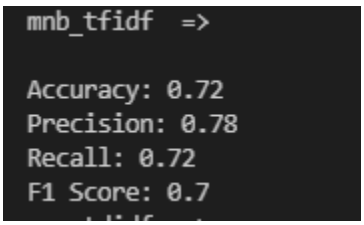
Result of SVM classifier and TFIDF weighted avg wordvector feature set

```
svm_tfidfwv  =>

Accuracy: 0.53
Precision: 0.54
Recall: 0.53
F1 Score: 0.51
```

Result of MNB classifier and bow feature set



Result of MNB classifier and TFIDF feature set



Result Interpretation

Comparing SVM and MB performance with related to bow and TFIDF feature set :

Here for bow feature set MNB model has performed better than SVM classifier also there is a significant improvement in precision of MNB model.

As for TFIDF feature set, SVM model has performed better compared to all classifier and feature set combination.

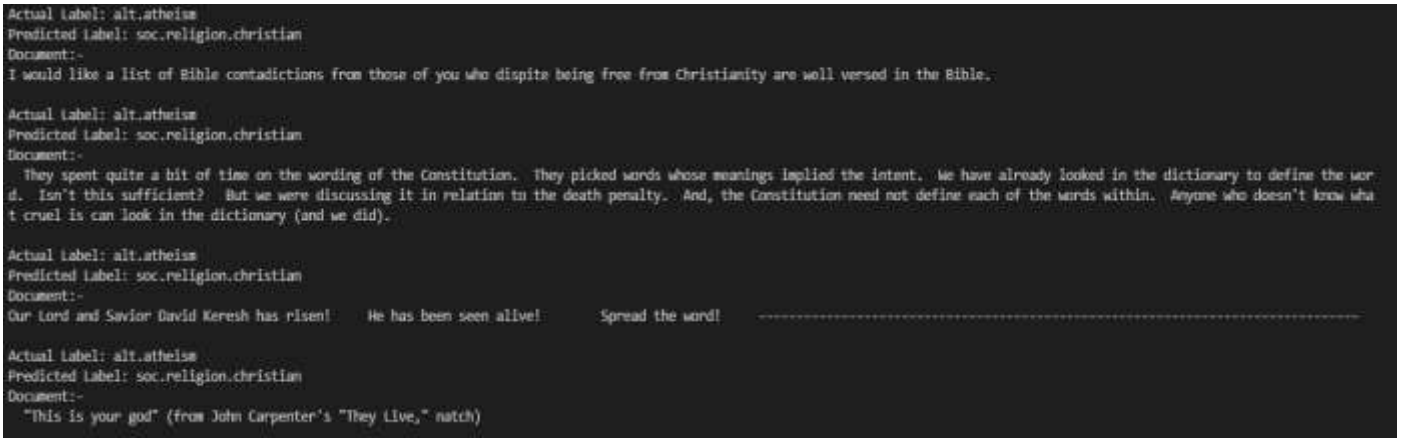
Overall MNB model has given high precision in both bow and TFIDF feature set compare to SVM classifier.

Comparing feature set bow, TFIDF, avg word vectors and TFIDF word vectors in SVM classifier

TFIDF feature set has given highest results compared to other features. It looks like for this case simple feature extraction methods like bow, TFIDF has given highest result compared to advanced technique like word2vec.

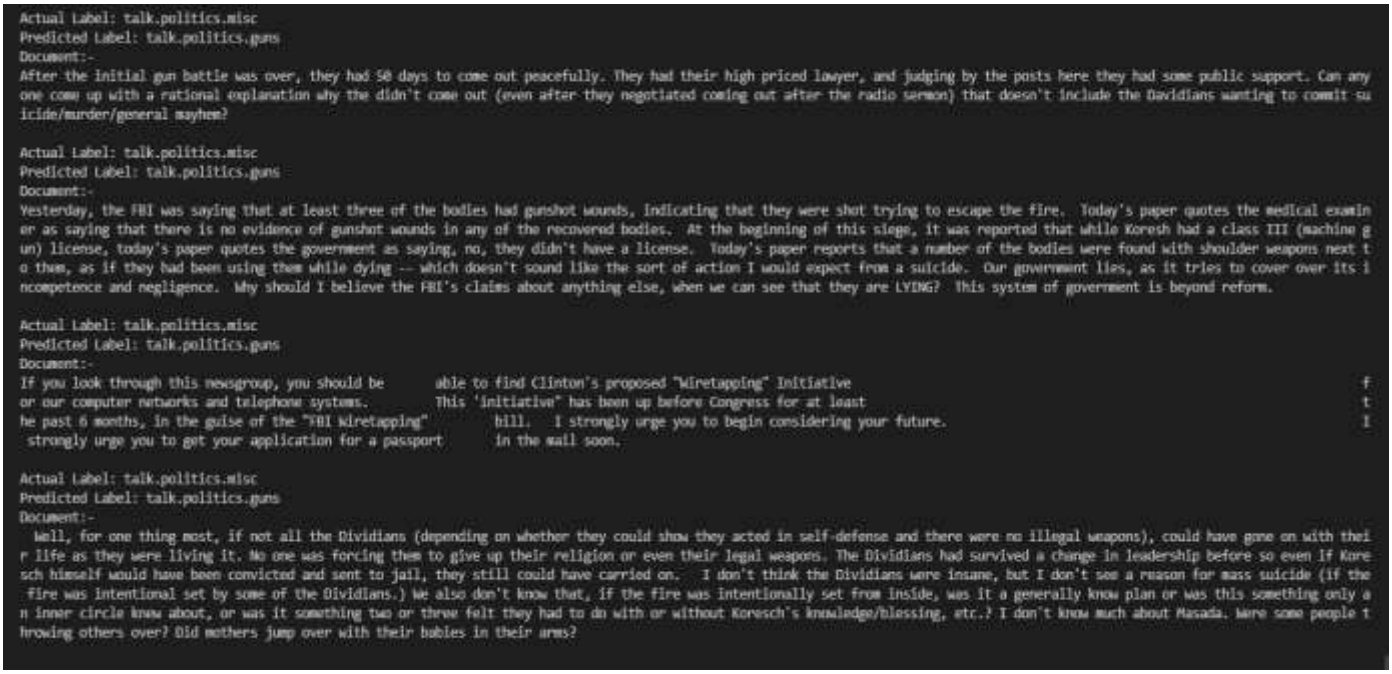
SVM TFIDF model Error Analysis Interpretation

Misclassified instances for Actual Label: alt.atheism ,Predicted Label: soc.religion.christian



Model identify document as Christian when there words like god, bible, death present. Having high weight for these word might be the case for misclassified instances.

Misclassified instances for Actual Label: talk.politics.misc Predicted Label: talk.politics.guns



Model tend to identify document as guns when ‘gun’,’ jail’ word present. This might be the case for misclassified instances.