Financial Text Sentiment Analysis using Supervised Learning Techniques

Name:	**Dinesh Adhithya**
Registration No./Roll No.:	18097
Institute/University Name:	IISER Bhopal
Program/Stream:	EECS
Problem Release date:	September 02, 2021
Date of Submission:	September 26, 2021

1 Introduction

The task of this work is to use supervised machine learning techniques to classify a financial statement from given corpus into positive, negative or neutral categories. A reduced form of data set [1] is used for this work. The data set contains 1807 unique words after removal of stop words and 1805 unique words upon further lemmatization of words in the sentences.The data set contains 456 sentences labelled as positive , 242 sentences labelled as negative and 1113 sentences labelled as neutral .The aim of this work is to use supervised learning models to classify sentences and use feature engineering and feature extraction techniques to improve performance of these models. Figure **??**.

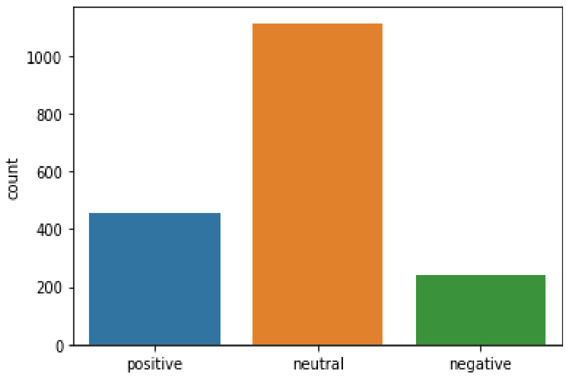

Figure 1: Plot showing frequency of occurrence of each class in data set

2 Methods

2.1 Sentiment WordNet Classifier

The sentences were tokenized and each word's part of speech was found using nltk [2] package.Then each word's positive sentiment and negative sentiment score was found and summed up for all words belonging to each sentence. A simple classifier was built using these scores to classify sentences.

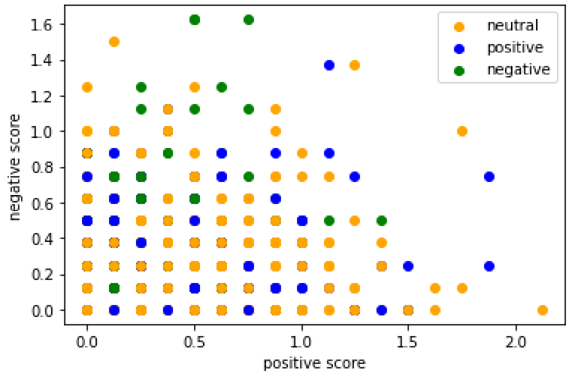

Figure 2: Scatter plot of positive and negative score predicted by SentiWordNet for data set

Table 1: Table showing Confusion Matrix of the SentiWordNet Classifier

	Labelled negative	Labelled neutral	Labelled positive
Predicted negative	65	154	23
Predicted neutral	95	660	358
Predicted positive	77	256	123

2.2 Sentiment Intensity Analyzer

This analyzer is also implemented in nltk [2] and returns the probability of a sentence being positive , negative or neutral in sentiment. The probability of sentence belonging to each label values was used to built a classifier.The confusion matrix of the classifier when tested on the data set.

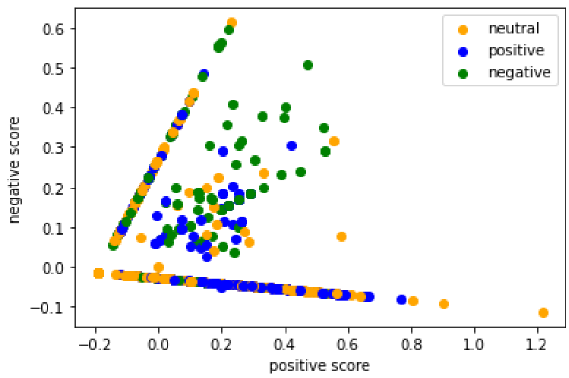

Figure 3: Scatter plot showing PCA 2 dimensional decomposition of class probabilities

Table 2: Table showing Confusion Matrix of the Sentiment Intensity Analyzer Classifier

	Labelled negative	Labelled neutral	Labelled positive
Predicted negative	2	239	1
Predicted neutral	1	1100	12
Predicted positive	0	438	18

2.3 Bag of Words Model

The data set which contains an array of sentences was tokenized using tfidf vectorizer , using Grid CV search best model parameters were selected , parameters the include manner in which input data is passed to machine learning models like range of n grams. Then parameters particular to the machine learning models was also optimized.Using Kfold split the training data was slit into 10 folds , the models were trained separately on each split and on average the best performing parameters pertaining to each model were selected.

The performance of various machine leaning models where explored such as Naive Bayes , Logistic Regression , Support Vector Machine , Random Forest Classifier and Decision Trees. The Naive Bayes classifier is a simple probabilistic model based on the Bayes rule , in this word a Multinominal Naive Bayes classifier[1] was used.Support Vector classifier[2] is based on constructing a decision boundary such that distance from the shortest points belonging to each label is maximised. Past works suggest us that a linear kernel is best suited for text classification[3].The logistic regression [4] works on a linear decision boundary to which a logistic function is applied , making sure the probability to which class a input belongs is given as output by the model. Although the logistic regression works well for two class classification problem but can also be used , when it predicts the probability a value belongs to a particular to a class by considering the next of the classes as another class. This is done for each class , the class for whom the probability is maximum is given as output.The decision tree classifier[5] works on simple decision rules based on data features. The random forest classifier[6] fits a number of decision trees on various sub samples of the data set and uses averaging to improve accuracy , this model works quite well in terms of reducing over fitting of data as the size of the data set is small.

The above models were implemented using sklearn[3] module in python language[4].

3 Evaluation Criteria

The performance of the classifiers are evaluated by using the standard precision, recall and f-measure. The precision and recall for two class classification problem can be computed as follows :

Metric	Formula
True positive rate, recall	$\frac{\text{TP}}{\text{TP+FN}}$
False positive rate	$\frac{\text{FP}}{\text{FP+TN}}$
Precision	$\frac{\text{TP}}{\text{TP+FP}}$
Accuracy	$\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$
F-measure	$\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$

Figure 4:

The closer the values of precision and recall, the higher is the f-measure. F-measure becomes 1 when the values of precision and recall are 1 and it becomes 0 when precision is 0, or recall is 0, or both are 0. Thus f-measure lies between 0 and 1. A high value of f-measure is thus desirable.

4 Analysis of Results

The SentiWordNet classifier largely is unable to precisely classify the sentences , the Sentiment Intensity Analyzer does better than SentiWordNet but largely don't work well as these models largely depend on the nature of the corpus they were trained on.The decision tree classifier performs poorly with macro average score of 0.62,whereas the random tree classifier being more robust by having a large number of decision trees in it , has an macro average score of 0.78 but performs poorly in recall

[1]https://scikit-learn.org/stable/modules/naive$_b$*ayes.html*
[2]https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html
[3]https://www.svm-tutorial.com/2014/10/svm-linear-kernel-good-text-classification/
[4]https://scikit-learn.org/stable/modules/generated/sklearn.linear$_m$*odel.LogisticRegression.html*
[5]https://scikit-learn.org/stable/modules/tree.html
[6]https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

values of negative and positive classes , hence there are a large number of false negatives in the negative and positive class data points.The best performing models happen to be support vector classifier and logistic regression. Their performance is almost indistinguishable with support vector classifier having a better recall score for positive class and Logistic regression has better recall score for negative class.The tables containing performances of various models is shown below.

Table 3: Table showing performance of the SentiWordNet Classifier

metric	precision	recall	f1-score	support
negative	0.27	0.27	0.27	242
neutral	0.62	0.59	0.60	1113
positive	0.24	0.27	0.26	456
accuracy	-	-	0.97	1115
macro avg	0.38	0.38	0.38	1811
weighted avg	0.48	0.47	0.47	1811

Table 4: Table showing performance of the Sentiment Intensity Analyzer Classifier

metric	precision	recall	f1-score	support
negative	0.67	0.01	0.01	242
neutral	0.62	0.99	0.76	1113
positive	0.58	0.04	0.07	456
accuracy	-	-	0.62	1811
macro avg	0.62	0.35	0.28	1811
weighted avg	0.62	0.62	0.49	1811

Table 5: Table showing performance of the Logistic Regression Classifier

metric	precision	recall	f1-score	support
negative	0.66	0.66	0.66	242
neutral	0.89	0.99	0.91	1113
positive	0.78	0.66	0.72	456
accuracy	-	-	0.83	1811
macro avg	0.77	0.75	0.76	1811
weighted avg	0.83	0.83	0.83	1811

Table 6: Table showing performance of the Naive Bayes Classifier

metric	precision	recall	f1-score	support
negative	0.63	0.48	0.54	242
neutral	0.89	0.92	0.89	1113
positive	0.78	0.66	0.67	456
accuracy	-	-	0.79	1811
macro avg	0.73	0.68	0.70	1811
weighted avg	0.78	0.79	0.79	1811

Table 7: Table showing performance of the Support Vector Classifier Classifier

metric	precision	recall	f1-score	support
negative	0.69	0.62	0.65	242
neutral	0.89	0.92	0.91	1113
positive	0.73	0.70	0.71	456
accuracy	-	-	0.83	1811
macro avg	0.77	0.75	0.76	1811
weighted avg	0.83	0.83	0.83	1811

Table 8: Table showing performance of the Decision Tree Classifier Classifier

metric	precision	recall	f1-score	support
negative	0.55	0.18	0.27	242
neutral	0.80	0.88	0.85	1113
positive	0.52	0.56	0.54	456
accuracy	-	-	0.71	1811
macro avg	0.62	0.54	0.55	1811
weighted avg	0.69	0.71	0.69	1811

Table 9: Table showing performance of the Random Forest Classifier Classifier

metric	precision	recall	f1-score	support
negative	0.76	0.46	0.57	242
neutral	0.85	0.96	0.90	1113
positive	0.72	0.64	0.68	456
accuracy	-	-	0.81	1811
macro avg	0.78	0.69	0.72	1811
weighted avg	0.80	0.81	0.80	1811

5 Discussions and Conclusion

Upon training on the best performing SVM model the following sentences were mislabelled:

Table 10: Table showing the mislabelled messages by SVM Model

Mislabelled Sentences first quarter Sacanfil net sale totalled EUR 50.0 operating profit EUR 4.7
Neste Shipping likely remain Finnish oil sector transport significant emergency supply
The growth net sale first half compared first half
Thus method cut working cost fasten planning building process
For net profit EUR million company paid dividend EUR 1.30 apiece
Operating profit totalled EUR 7.0 loss EUR 4.0 second quarter
The Group profit tax EUR 0,2 7,8 million
Previously delivered custom solution Electronics making commercially available mobile terminal vendor
Operating profit EUR 11.4 EUR 7.5
The company net profit amounted 55.5
Arvo Vuorenmaa Loviisa plant application new procedure quite confident approval granted
Very recommendable Nokian according automobile ADAC Operating profit totaled EUR 5.5 EUR -0.7
With appointment Kaupthing Bank aim co-ordinate Capital Markets activity within Group improve
The number collection error fell considerably operation speeded EBIT margin 1.4 5.1
The group posted net sale 35.3 mln euro 46.5 mln operating profit 760,000 euro 1.0 mln
Operating profit margin increased 11.2 11.7
Operating profit totalled EUR 21.1 EUR 18.6 representing 9.7 net sale

The mislabelled sentences quite a few of them have numbers and the potential meaning of such numbers usually money taken in context with sentiment of words around them also should be taken into consideration in future models used for financial text classification. A few neutral statements have been wrongly classified are largely due to small training set and the vocabulary not seeing sufficient words and larger data set can be used in the future works.

The exploration of models such as LSTM , GRU which take context into consideration , as many words in the English language's sentiment depends on the context and words around it.

The above models were implemented in jupyter notebooks [5] using the scikit- learn package [3] in python language. The book by Witten , Frank and Hall served as a good reference [6]. Stack exchange has also served as a good resource for debugging Latex and Python scripts. All the codes used for this work can be found at the GitHub repository. Any changes you wish to suggest can be reported on GitHub itself.

References

[1] P. Malo, Ankur Sinha, P. Korhonen, J. Wallenius, and P. Takala. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65, 2014.

[2] Edward Loper and Steven Bird. Nltk: The natural language toolkit. *CoRR*, cs.CL/0205028, 2002.

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[4] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual.* CreateSpace, Scotts Valley, CA, 2009.

[5] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing. Jupyter notebooks – a publishing format for reproducible computational workflows.

[6] I. H. Witten, E. Frank, and M. Hall. *Data Mining: Practical Machine Learning Tools and Techniques.* Elsevier, third edition, 2011.