Social Informatics Business, Politics, Law, Environmental Sciences & Technology Journal

Vol 8, No 2, 2022 Series

Sentiment Analysis-Based Student Performance Predictor (SPP)

Sarumi, Jerry. Abayomi[1] (PhD), Shomope Adewale A[2]
Department of Computer Science
Lagos State University of Science & Technology
Ikorodu, Lagos State, Nigeria
E-mail: jerrytechnologies@yahoo.co.uk, shomopelaspocomputer@gmail.com
Phone: +2348023408122

ABSTRACT

We implement a student performance predictor (SPP) using sentiment analysis-Based statistical modelling. In the first step (data ingestion) the data is collected and stored in a database. After collecting the data, the data is cleaned and/or transformed. The data is divided into two sets: a training set and a testing set. In the next step a mathematical model is built based on the training set and then the model will be tested against the testing set. In order to improve the results, the user can make decision about creating or choosing different data and feature vectors (data presentation style), after results are produced from the model.

Keywords: Sentiment Analysis, Students Performance, Predictor, Ingestion, Data, Training.

Journal Reference Format:
Aweh O., Parker, E-C. A. & Dagah, W.D. (2022): Sentiment Analysis-Based Student Performance Predictor (SPP).
Social Informatics, Business, Politics, L:aw, Environmental Sciences & Technology Journal. Vol. 8, No. 2
Pp 65-78. www.isteams/socialinformaticsjournal

1. INTRODUCTION

There are three categories of machine learning that are based on their nature of learning. Data collection and dataset creation is the first step when you want to create a statistical model using machine learning. The dataset is commonly divided into three subsets: a training set, a validation set and a test set. The training set is used to train the statistical model, the validation set is used to estimate how well the model is trained and the test set is used to measure the performance of the model. The student performance predictor (SPP) is a prototype tool for prediction of the student academic performance using machine learning. The flow of how SPP is used is shown below.

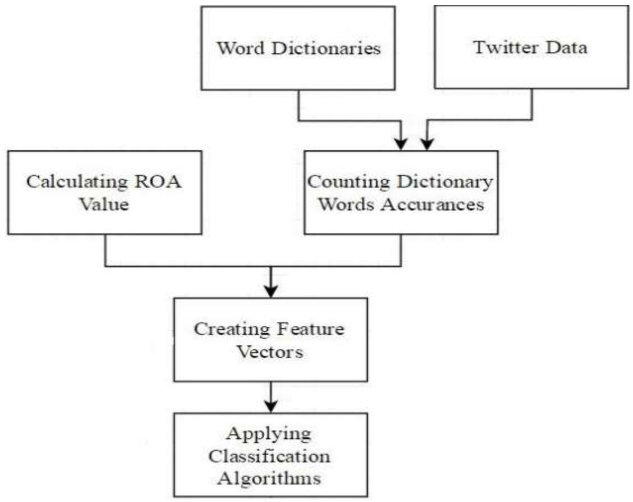

Figure 1: Steps toward financial prediction

The first step is to collect relevant data, in this thesis we use data from Twitter. In order to detect the sentiment of a tweet or a group of tweets, we use the bag of word method. The bag of word method focus on the words or in some cases set of words (a string of words), regardless of the context of sentence. We use a list of words (from a dictionary) and all words that are attached to a sentiment. The words are either positive or negative. In the experiment we have used two different dictionaries one with that is developed for financial purposes and one more general. The second step is to count the number of occurrence of each word present in the dictionaries in the extracted tweets. The result is combined with the ROA for the corresponding time period and included in the feature vectors. In the forth step machine learning algorithms will be applied on the feature vectors to train a model to predict if the ROA increases or decreases based on the sentiment of the tweets. The classification algorithms that we have used to train the model are Random Forest, Naive Bayes and Adaboost.

Student Performance Predictor (SPP)

Predictor implementation various programming languages and tools are used in the implementation of the SPP.

Collecting data

In order to download tweets a web scraper is written in python programming language. At the first step a web search query will be made by a python library called selenium. In the second step the HTML contents will be stored to driver's page source of a web browser. In the third step a python library called beautifulsoup is used to organize and extract the required data from the HTML source. At the last step the tweets will be saved as a comma separated version (CSV) file and then stored in a MySQL database to ease the data management.

Feature vectors creation

In this project a program for creating feature vectors is written in Java. The program uses the word dictionaries and count the number of occurrence of each dictionary word in the tweets. The result is stored in a vector. The class variable it the company's performance. The value of class variable is 1 in case of over-performance and 0 in case of under-performance. The format of a feature vector is shown in Figure 1.

Table 1: The format of a feature vector.

Positive Word Counts	Negative Word Counts	Positive	Neutral	Negative	# of Likes	# of Retweets	Class Variable
X1…X45…	X560…X1980…	X2500	X2501	X2502	X2503	X2504	Y

2. DEVELOPMENT TECHNOLOGY

Programming Language (Python)

In order to download tweets a web scraper is written in python programming language. At the first step a web search query will be made by a python library called selenium. In the second step the HTML contents will be stored to driver's page source of a web browser. In the third step a python library called beautifulsoup is used to organize and extract the required data from the HTML source. At the last step the tweets will be saved as a comma separated version (CSV) file and then stored in a MySQL database to ease the data management.

Development Tools (Weka)

All experiments are done using Weka. Weka has a collection of data mining algorithms, predictive modeling and tools for visualization and a graphical user interface for ease of access to its functions. Three different classification algorithms are used in our experiments: Random forest, Naive Bayes and AdaBoost. Information Gain feature selection method is been used for Naive Bayes classifier.

Table 2: Sample words from the two different dictionaries

Regular dictionary	Example
Positive Emotions	happy, pretty, good
Negative Emotions	hate, worthless, enemy, hurt
Financial dictionary	**Example**
Positive Emotions	best, achieve, able
Negative Emotions	abandoned, misprice, untrusted

For data balancing, the SMOTE algorithm and Weka Randomize filter are used. The default settings for each algorithm in Weka are:

- Random Forest: Number Of Trees: 100, Seed = 1
- AdaBoost: Number of Iteration = 10, Seed = 1, Weight Threshold = 100.
- SMOTE: Nearest Neighbor = 5, Percentage (percentage of SMOTE instances to create) = 100, Random seed =1

Linguistic Inquiry and Word Count (LIWC)
LIWC (Linguistic Inquiry and Word Count) is a text analysis program. It processing feature is the program itself, which opens a series of text files—which can be essays, poems, blogs, novels, and so on—and then goes through each file word by word. It calculates the percentage of words in a given text or tt calculates the degree to which (LIWC).

The second dictionary (called the f inancial dictionary) is called Loughran-McDonald master dictionary. The Loughran-McDonald master dictionary is an extension of the 2of12inf wordlist that includes an addition of the words that are appearing in companies annual reports. The 2of12inf is a wordlist from SCOWL (Spell Checker Oriented Word Lists) and Friends consisting of English words that are useful for creating high-quality list of words for spell checkers.

3. IMPLEMENTATION

The implementations are done in Python. S cikit-learn is used for feature representation, classification, similarity measures and evaluation purpose. Natural Language T oolkit(Nltk) is used for stemming and stop word removal in data preprocessing. Pandas is used for handling dataset. NumPy is used to handle multi-dimensional arrays.

Datasets Evaluation
There are open repositories which are maintained by researchers to keep an up-to-date list of currently available datasets and hyperlinks. The datasets are available online and are extracted from the World Wide Web. These are the datasets commonly used for this study, they are open source and freely available online. The proposed system was tested on the following datasets collected from Twitter pertaining to different topics:

A. Stanford - Sentiment140 corpus: Sentiment140 dataset (Go et al., 2019) is generally used to train and test the system. This consists of 1,600,000 training tweets with eight lakh tweets labelled positive label and eight lakh labelled negative.

B. Health Care Reform (HCR): This dataset (Speriosu et al., 2016) was assembled by searching tweets with the #hcr. The tweets with positive sentiment and negative sentiment are considered for the experiment. This dataset consists of 888 tweets (365 positive and 523 negative).

C. First GOP debate twitter sentiment dataset: This dataset from Crowdflower consists tweets on the first GOP debate for the 2016 presidential nomination. This GOP debate dataset consists of 13871 tweets with the positive, negative or neutral sentiment. Neutral sentiment tweets were not taken for research. So final dataset contains 10729 tweets with 2236 positive and 8493 negative labels.

D. Twitter sentiment analysis dataset: This dataset has 99989 training tweets. Each tweet is either positive or negative. This dataset consists of 43532 negative and 56457 positive tweets. This dataset is available at kaggle.

5. EXPERIMENTS
In this section the experimental setup along with the results are described.

Table 4.3: The Datasets Used In The Experiments

Dataset	Description	Size	Time period
TW_{BMW}	Tweets related to BMW	677596	2007-2015
TW_{VW}	Tweets related to Volkswagen	151648	2012-2015

Table 4: Company's performance based on the ROA.

Year	Quarter	BMW	Volkswagen
2015	Quarter 1	Over-perform	Under-perform
	Quarter 2	Over-perform	Over-perform
	Quarter 3	Under-perform	Under-perform
2014	Quarter 1	Over-perform	Under-perform
	Quarter 2	Over-perform	Over-perform
	Quarter 3	Under-perform	Under-perform
2013	Quarter 1	Under-perform	Under-perform
	Quarter 2	Over-perform	Over-perform
	Quarter 3	Under-perform	Under-perform
2012	Quarter 1	Over-perform	Under-perform
	Quarter 2	Over-perform	Under-perform
	Quarter 3	Under-perform	Over-perform
2011	Quarter 1	Over-perform	—
	Quarter 2	Over-perform	—
	Quarter 3	Over-perform	—
2010	Quarter 1	Over-perform	—
	Quarter 2	Over-perform	—
	Quarter 3	Over-perform	—
2009	Quarter 1	Under-perform	—
	Quarter 2	Over-perform	—
	Quarter 3	Under-perform	—
2008	Quarter 1	Under-perform	—
	Quarter 2	Over-perform	—
	Quarter 3	Under-perform	—
2007	Quarter 1	Over-perform	—
	Quarter 2	Over-perform	—
	Quarter 3	Over-perform	—

Dataset

Two datasets are used for the experiments. The first dataset denoted as TWBMW contains tweets where BMW is either mentioned or used in a hashtag (#BMW). The second dataset is called TWV W contains tweets where Volkswagen is either mentioned or used in a hashtag (#Volkswagen). The two datasets are described in Table 4.2

An example of a negative tweet from TWBMW is:
"BMW is ruining the M-division brand by releasing crap like the "X6 M"
- http://tinyurl.com/cb2nq7"

An example of a positive tweet from the same dataset is:
"Track drive reveals excellent balance of the 2015 BMW 228i - Torque
News http://bit.ly/1xk4xj7 - #BMW"

An example of a neutral tweet (neither positive or negative) from the
same dataset:
"mclaren should come back later in the race when ferrari and bmw have
to use the hard tyres hopefully, anyway"

The sentiment of each tweet is determined by counting the occurrence of positive and negative words. If a tweet contain more positive words than negative words, the sentiment is considered positive, if there are more negative words than positive words, the sentiment is considered negative. If a tweet contain the same amount of positive and negative words the sentiment is considered to be neutral.

Dictionaries

We have used two different dictionaries to determine the sentiment of tweets. The first dictionary (called the regular dictionary) is inspired by the positive and negative emotions from the tool Linguistic Inquiry and Word Count.

Quarterly Reports

To obtain the value on return on asset (ROA) for each quarter, BMW quarterly reports (10-Q reports) are downloaded from (https://www.bmwgroup.com/en/ investor-relations/financial-reports.html) and Volkswagen quarterly reports are downloaded from (http://quicktake.morningstar.com/ stocknet/secdocuments.aspx?symbol=vlkay). The value of ROA is not explictly mentioned in the quarterly reports and therefore it is calculated manually using the value of the total income and and the total assets value. In Table 4 performance of BMW and Volkswagen in different quarter of the year is shown.

Experiments Result

The performance of the proposed ensemble classifier is compared with the individual traditional classifier and majority voting ensemble classifier. The results are shown in Table 1. Stanford - Sentiment140 corpus consist of 1.6 million tweets. Bakliwal et al., 2016 Go et al., 2018 Sperioosu et al., 2019 and Prusa et al., 2020 are also used this dataset to evaluate their system. Due to the computational limitation of system, it is very difficult to test the proposed system with 1.6 million tweets. Therefore, only 1,00,000 tweets are used for experiments as sampling which is only 6.25% of total tweets to test the proposed system. Over 1,00,000 tweets, 70,000(70%) tweets are used for training the system and 30,000(30%) are used for testing the system. The results show that the proposed ensemble classifier performed better than stand-alone classifier and majority voting ensemble classifier on different types of datasets.

We have done four different experiments to get an understanding on the possibilities to predict a company's performance based on public opinion extracted from social media. The experiments are different in terms of the number of feature vectors used, the features and the choice of classifier. All experiments have the same classifier setup. For each relevant time period, a number of feature vectors are created from the datasets. For each time period a variable describing if the company was under-performing or over performing (relative to previous quarter) is added. The differences between the experiments are the number of feature vectors that are created for the dataset and what dictionary that is used. The results for the different classifiers are described as confusion matrices in which we present the number of true positives, false negatives, true negatives, and false positives as illustrated in Table 5

Table 5: Confusion Matrix

	Predicted class	
Actual class	**True Neg. (TN)**	**False Pos. (FP)**
	False Neg. (FN)	**True Pos. (TP)**

To evaluate the results we use the measures accuracy, precision, recall and F-score that can be derived from the confusion matrix. Accuracy is defined as:

Accuracy is defined as:

$$\frac{TP + TN}{TP + FP + TN + FN}$$

precision is defined as:

$$\frac{TP}{TP + FP}$$

recall as:

$$\frac{TP}{TP + FN}$$

and F-score (to measure test's accuracy) as:

$$\frac{2 * precision * recall}{precision + recall}$$

Table 6: The results for experiment using dataset

Dataset	Classifier	Over-perform	Under-perform	Accuracy	Precision	Recall	F-Score
TW_{VW}	Random Forest	604	194	86.17%	0.953	0.757	0.842
		30	792				
TW_{VW}	Naive Bays	567	231	77.22%	0.804	0.711	0.752
		138	684				
TW_{VW}	AdaBoost	448	350	60.86%	0.612	0.561	0.584
		284	538				

The data set holds 16488 tweets. Each tweet contains a statement regarding a university or more from the TU9 in Germany. For the training set, 5000 tweets were chosen randomly and got annotated manually by one sentiment either a “Positive” tweet or “Not Positive” tweets. From the 5000 tweets, 4000 tweets where chosen randomly divided equally between 2000 “Positive” and 2000 “Not Positive” tweets. They are the input for the training step of the Naive Bayes classifier (see Figure 1). The results section evaluates three main aspects of the presented method:

a) Measuring the classifier efficiency based on the suggested filtering and features ex-traction steps.
b) Establishing a comparison between the TU9 based on each university’s tweets trying to prove the hypothesis that social media content may act as an indicator for university comparisons.
c) Investigating the tweets sentiment on daily basis for each university to obtain feedback on different events and activities. Each is presented in the following sections.

Table 7.

Cross comparison of the results obtained from base classifiers, majority voting ensemble and proposed ensemble classifier. Pre, Rec and F1 refer to the Precision, Recall and F-measure

Techniques	Accuracy(%)	Positive class			Negative class			Average
		Pre(%)	Rec(%)	F1(%)	Pre(%)	Rec(%)	F1(%)	F1(%)
Stanford- Twitter Sentiment Corpus								
Naive Bayes	75.19	75.63	74.47	75.05	74.76	75.91	75.33	75.19
Random Forest	71.76	67.78	83.18	74.70	78.12	60.30	68.06	71.38
Support Vector Machine	75.61	73.98	79.18	76.49	77.50	72.03	74.67	75.58
Logistic Regression	74.15	72.37	78.30	75.21	76.25	69.98	72.98	74.09
Majority Voting	74.80	71.43	82.83	76.71	79.47	66.73	72.55	74.63
Proposed Ensemble	75.81	74.80	78.00	76.36	76.91	73.61	75.22	**75.79**
Health Care Reform Dataset								
Naive Bayes	71.80	59.37	61.29	60.32	78.82	77.46	78.13	69.22
Random Forest	71.43	67.35	35.48	46.48	72.35	90.75	80.51	63.49
Support Vector Machine	70.30	56.86	62.36	59.49	78.66	74.57	76.56	68.02
Logistic Regression	69.92	65.85	29.03	40.30	70.67	91.91	79.90	60.10
Majority Voting	71.05	66.00	35.48	46.15	72.22	90.17	80.20	63.17
Proposed Ensemble	73.68	63.85	56.99	60.23	78.14	82.66	80.34	**70.28**
First GOP Debate Dataset								
Naive Bayes	82.20	57.48	69.24	62.82	90.95	85.79	88.30	75.56
Random Forest	82.57	85.20	23.89	37.32	82.40	98.85	89.88	63.60
Support Vector Machine	83.44	62.17	60.66	61.40	89.16	89.76	89.46	75.43
Logistic Regression	81.51	83.77	18.45	30.25	81.40	99.01	89.35	59.80
Majority Voting	82.60	85.64	23.89	37.36	82.41	98.89	89.90	63.63
Proposed Ensemble	85.83	73.59	54.22	62.44	88.16	94.60	91.27	**76.85**
Twitter Sentiment Analysis Dataset								
Naive Bayes	73.65	77.96	77.18	77.57	67.57	68.56	68.06	72.81
Random Forest	70.61	68.73	92.13	78.73	77.73	39.58	52.45	65.59
Support Vector Machine	74.36	76.06	82.56	79.18	71.33	62.55	66.65	72.91
Logistic Regression	73.44	73.89	85.08	79.09	72.49	56.68	63.61	71.35
Majority Voting	73.83	72.99	88.36	79.94	75.92	52.88	62.34	71.14
Proposed Ensemble	74.67	76.62	82.18	79.30	71.31	63.85	67.37	**73.33**

social Informatics Journal
Business, Politics, Law, Environmental Sciences & Technology
Vol 8, No 2, 2022 Series

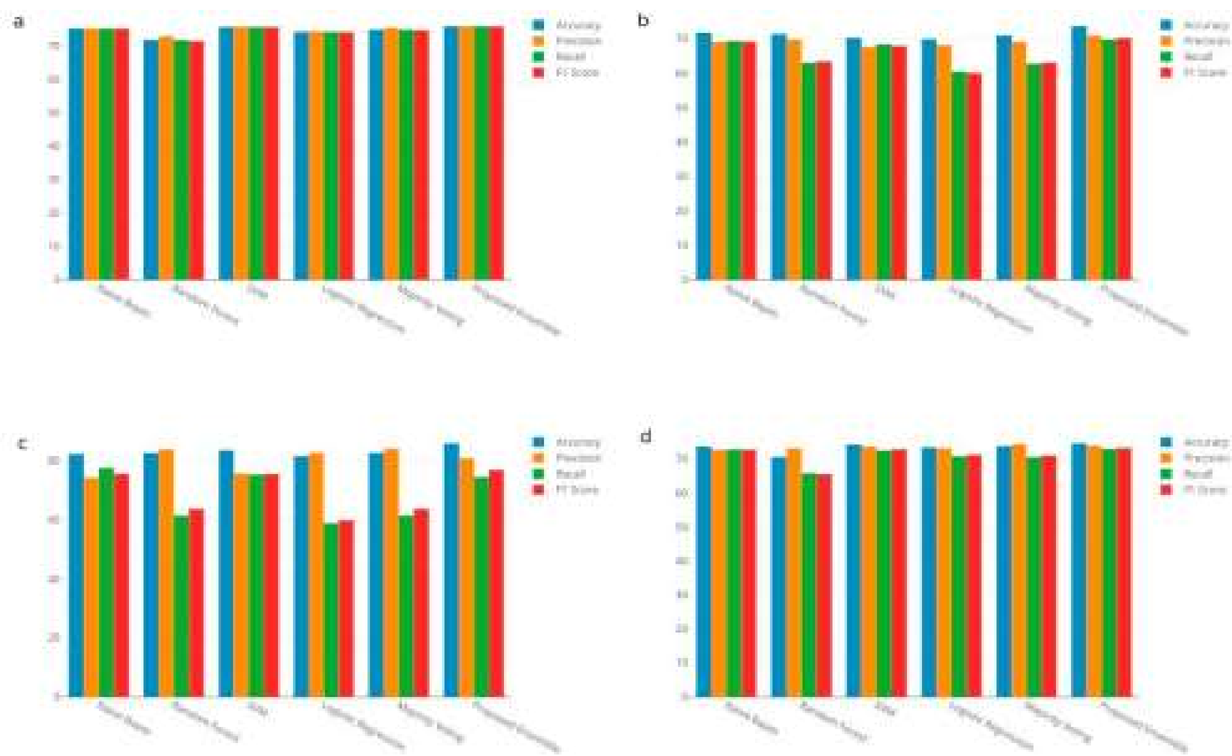

Figure .8: Performance Evaluation of Different Base Classifiers

Fig. 4.3 Performance evaluation of different classifiers with (a) Stanford dataset; (b) HCR dataset; (c) GOP debate dataset; (d) Twitter sentiment dataset.

Evaluation metrices are illustrated as (Han, 2015)

$$\text{Recall} = \frac{\text{T rue Pos Sentiment}}{\text{T rue Pos Sentiment + False Neg Sentiment}}$$

$$\text{Precision} = \frac{\text{T rue Pos Sentiment}}{\text{T rue Pos entiment + False Pos Sentiment}}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision + Recall}}$$

$$\text{Accuracy} = \frac{\text{True Pos Sentiment + T rue Neg entiment}}{\text{True Pos Sentiment + False Neg Sentiment + False Pos Sentiment + True Neg Sentiment}}$$

6. DISCUSSIONS

In the first experiment one feature vector was created for each quarter of the year, which means 27 data instances in total. Low number of data instances can be one of the reasons that the accuracy is lower in compare to other experiments. In the second experiment, instead of counting number of words and use them as features, the differences of word counts from previous quarter is used and the prediction accuracy has dropped for random forest algorithm while it showed a little improvement in other classifiers. The reason for getting low accuracy with random forest classifier could be that the sentiment in feature vectors should not be created in relation to other feature vectors. In the third and forth experiment, one feature vector is created per 100 tweets and the datasets are balanced, then the prediction accuracy improves. This could be due to balanced number of instances.

Among all of the experiments that is done, except experiment 2, the most accurate classifier was Random forest classification algorithm, from the third experiment which provided 86.17% accuracy in an experiment where 100 tweets from TWV W dataset were combined into one feature vector and the regular dictionary was used as features.

The best results was obtained when using random forest. Random forest ranks the variables in the feature vector, and also relation between each variables while splitting nodes, in order to produce higher accuracy. The data used to train the random forest classifier was balanced and therefore a more accurate classification model could be produced.

The rest of the tweets have been classified by the learned classifier. Each tweet belongs to one sentiment class. Tweets were divided over the TU9, showing how many “Positive” and
polytechnic name. The results might act as an indicator on how the higher education environment at each university is perceived by Twitter users. Such indicator which is supported by the social media content can play a role in enhancing the universities rankings.

7. CONCLUSION

In the area of twitter sentiment analysis (SA), the major approach is to compare the different base classifiers and select the best among them to implement tweet SA. The ensemble classification techniques have been widely used in many areas to solve the classification problem. But in case of tweet sentiment analysis, comparatively little work has been done on the use of ensemble classifiers. This research provides methodology and social media tools for performing sentiment analysis on posts containing information about the institution shared on twitter, and machine learning algorithms that can be used to predict the schools performances.

BIBLIOGRAPHY

1. Reza Zafarani, Mohammad Ali Abbasi, Huan Liu (2019). Social Media Mining Cambridge University
2. Buchem, I., Ebner, M., Schön, S., Appelt, R., & Kaiser, S. (2018). Blogging und Microblogging. Lehrbuch für Lernen und Lehren mit Technologien: 2. Auflage (2013), 239.
3. Gapski, H., & Gräßer, L. (2017). Medienkompetenz im web 2.0–lebensqualität als zielperspektive. Praxis Web, 2, 11-34.
4. Hisserich, J., & Primsch, J. (2020). Wissensmanagement in 140 Zeichen: Twitter in der Hochschullehre. Community of Knowledge.
5. Maynard, D., & Funk, A. (2016). Automatic detection of political opinions in tweets. In Extended Semantic Web Conference (pp. 88-99). Springer, Berlin, Heidelberg.
6. Go, A., Bhayani, R., & Huang, L. (2019). Twitter sentiment classification using distant supervision. CS224N project report, Stanford, 1(12), 2009.
7. Atefeh, F., & Khreich, W. (2015). A survey of techniques for event detection in twitter. Computational Intelligence, 31(1), 132-164.
8. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. J. (2016). Sentiment analysis of twitter data. In Proceedings of the workshop on language in social media (LSM 2011) (pp. 30-38).
9. Kouloumpis, E., Wilson, T., & Moore, J. (2016). Twitter sentiment analysis: The good the bad and the omg!. In Fifth International AAAI conference on weblogs and social media.
10. Pak, A., & Paroubek, P. (2020). Twitter as a corpus for sentiment analysis and opinion mining. In LREc (Vol. 10, No. 2010, pp. 1320-1326).
11. Davidov, D., Tsur, O., & Rappoport, A. (2020). Enhanced sentiment learning using twitter hashtags and smileys. In Coling 2010: Posters (pp. 241-249).
12. Kaplan, A. M., & Haenlein, M. (2020). Users of the world, unite! The challenges and opportunities of Social Media. Business horizons, 53(1), 59-68.
13. Subhabrata Mukherjee. (2017). Sentiment analysis. Indian Institute of Technology, Bombay Department of Computer Science and Engineering.
14. Hagel, J. I. I. I., Brown, J. S., & Davison, L. (2020). The best way to measure company performance. Harvard Business Review, 4(6), 56-60.
15. Needles, B. E., Powers, M., & Crosson, S. V. (2018). Principles of accounting. Cengage Learning.
16. Granger, C. W. (2019). Investigating causal relations by econometric models and cross-spectral methods. Econometrica: journal of the Econometric Society, 424-438.
17. Michael T. Lash and Kang Zhao (2016). Early Predictions of Movie Success: the Who, What, and When of Profitability Artificial Intelligence (cs.AI); Social and Information Networks (cs.SI).
18. Sahar Nassirpour, Parnian Zargham, Reza Nasiri Mahalati (2017). Electronic Devices Sales Prediction Using Social Media Sentiment Analysis Project report Stanford university.
19. Richard Socher Jeffrey Pennington, Eric H. Huang Andrew, Y. Ng Christopher, D. Manning (2016). Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions Proceedings of the Conference on Empirical Methods in Natural Language Processing Pages 151- 161

20. Tristan Fletcher, Fabian Redpath and Joe DAlessandro (2019) Machine Learning in FX Carry Basket Prediction Proceedings of the International Conference of Financial Engineering, vol. 2, page 1371-1375.
21. Tim Loughran and Bill McDonald (2016) When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks The Journal of Finance, Vol. LXVI, NO. 1
22. Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining.In LREc (Vol. 10, No. 2010, pp. 1320-1326).
23. Douglas M. McNair, Maurice Lorr, and Leo F. Droppleman (2021). Manual for the Profile of Mood States San Diego, CA: Educational and Industrial Testing Service
24. Carol McDonald. (2015). Parallel and Iterative Processing for Machine Learning Recommendations with Spark https://www.mapr.com/blog/ parallel-and-iterative-processing-machine-learning-recommendations-spark
25. Karina Gibert, Miquel Snchez-Marr, Vctor Codina. (2020). Principles of Accounting. International Environmental Modelling and Software Society (iEMSs).
26. Rokach, L., & Maimon, O. Z. (2017). Data mining with decision trees: theory and applications (Vol. 69). World scientific.
27. Freund, Y., & Schapire, R. E. (2016). Experiments with a new boosting algorithm. In icml (Vol. 96, pp. 148-156).
28. Sylvain Arlot, (2019). A survey of cross-validation procedures for model selection. Journal of Machine Learning Research , p1089-1105.
29. Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, W. Philip Kegelmeyer. (2017). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, page 321357.
30. Speriosu, M., Sudan, N., Upadhyay, S., & Baldridge, J. (2016). Twitter polarity classification with label propagation over lexical links and the follower graph. In Proceedings of the First workshop on Unsupervised Learning in NLP (pp. 53-63).
31. Medhat, W., Hassan, A., & Korashy, H. (2019). Sentiment analysis algorithms and applications:
32. A survey. Ain Shams engineering journal, 5(4), 1093-1113.
33. Davidov, D., Tsur, O., & Rappoport, A. (2020). Enhanced sentiment learning using twitter hashtags and smileys. In Coling 2010: Posters (pp. 241-249).
34. Read, J. (2015). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In Proceedings of the ACL student research workshop (pp. 43-48).
35. Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., & Liu, B. (2016). Combining lexicon-based and learning-based methods for Twitter sentiment analysis. HP Laboratories, Technical Report HPL-2011, 89.
36. Onan, A., Korukoğlu, S., & Bulut, H. (2016). A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification. Expert Systems with Applications, 62, 1-16.
37. Yoon, V. Y., Hostler, R. E., Guo, Z., & Guimaraes, T. (2018). Assessing the moderating effect of consumer product knowledge and online shopping experience on using recommendation agents for customer loyalty. Decision Support Systems, 55(4), 883-893.
38. Lin, J., & Kolcz, A. (2017). Large-scale machine learning at twitter. In Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data (pp. 793-804).
39. Rodriguez-Penagos, C., Atserias, J., Codina-Filba, J., García-Narbona, D., Grivolla, J., Lambert, P., & Saurí, R. (2018). Fbm: Combining lexicon-based ml and heuristics for social media polarities. In Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2018) (pp. 483-489).
40. Hassan, A., Abbasi, A., & Zeng, D. (2018). Twitter sentiment analysis: A bootstrap ensemble framework. In 2018 international conference on social computing (pp. 357-364). IEEE.

41. Clark, S., & Wicentwoski, R. (2018). Swatcs: Combining simple classifiers with estimated accuracy. In Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2018) (pp. 425-429).
42. Jianqiang, Z. (2016). Combing semantic and prior polarity features for boosting twitter sentiment analysis using ensemble learning. In 2016 IEEE first international conference on data science in cyberspace (DSC) (pp. 709-714). IEEE.
43. Pang, B., Lee, L., & Vaithyanathan, S. (2017). Thumbs up? Sentiment classification using machine learning techniques. arXiv preprint cs/0205070.
44. Onan, A., Korukoğlu, S., & Bulut, H. (2016). A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification. Expert Systems with Applications, 62, 1-16.
45. Prusa, J., Khoshgoftaar, T. M., & Dittman, D. J. (2015). Using ensemble learners to improve classifier performance on tweet sentiment data. In 2015 IEEE International Conference on Information Reuse and Integration (pp. 252-257). IEEE.
46. Coletta, L. F., da Silva, N. F., Hruschka, E. R., & Hruschka, E. R. (2019). Combining classification and clustering for tweet sentiment analysis. In 2014 Brazilian conference on intelligent systems (pp. 210-215). IEEE.