

A Comparison of Machine Learning Techniques for Sentiment Analysis

Shahzad Qaiser¹, Nooraini Yusoff^{2*}, Ramsha Ali³, Muhammad Akmal Remli⁴, Hasyiya Karimah Adli⁵

¹Department of Computer Science, Capital University of Science and Technology (CUST) Islamabad Expressway, Kahuta Road Zone-V, Islamabad, Pakistan

³School of Quantitative Sciences, UUM College of Arts and Sciences, Universiti Utara Malaysia, 06010 UUM Sintok, Kedah, Malaysia

^{2,4,5}Institute for Artificial Intelligence and Big Data(AIBIG), Universiti Malaysia Kelantan, City Campus, 16100 Kota Bharu, Kelantan, Malaysia

Corresponding author: ²*nooraini.y@umk.edu.my

Article History: Received: 10 November 2020; Revised: 12 January 2021; Accepted: 27 January 2021; Published online: 05 April 2021

Abstract: The availability of the data has increased tremendously due to the excess usage of social media platforms like Twitter and Facebook. Due to the abundant availability of data, scientists, businesses, educationalists and other people working under different roles have started using Sentiment Analysis (SA) to get in-depth knowledge about the sentiments of the people regarding any topic of interest. There are many techniques to implement SA, and one of them is Machine Learning (ML). This study is focused on the comparison of ancient ML methods such as Naïve Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), and a modern method, i.e., Deep Learning (DL). The ML techniques are applied to a single dataset to compare their performance in terms of accuracy to understand how they perform against each other. The study found that DL performed the best with 96.41% accuracy followed by NB and SVM with 87.18% and 82.05% respectively. DT performed the poorest with 68.21% accuracy.

Keywords: facebook, twitter, sentiment analysis (SA), machine learning (ML), deep learning (DL), decision tree (DT), dataset

1. Introduction

In this era, internet usage has been drastically increased [1]–[6]. Today, the majority of people have access to social media platforms such as Twitter, Facebook, and Instagram. People use social media for various purposes such as connecting to the family and friends, getting news updates, or advertising their businesses. When people use social media, they share various kinds of posts with the audience. Such posts can have essential pieces of information such as someone’s opinion regarding a newly launched product by a famous brand or someone’s sentiment regarding how he feels about a newly launched robot to replace humans at work. Such data from social media platforms can hint towards the opinions and sentiments of the people regarding any particular subject at any specific place which can help study human opinion or sentiments and take counter-measures if needed.

Sentiment Analysis (SA) can be performed on social media data to study human opinion or sentiment regarding any subject [7]. SA can be used to extract the user’s opinions or sentiments and categorize them into three classes, i.e., {positive, neutral or negative} where “Positive” means the person holds positive opinion or sentiment regarding the discussed subject while “Neutral” means the person is not “Positive” nor “Negative” and “Negative” means the person holds a negative opinion or sentiment. [1], [2], [8]. There are multiple techniques to apply SA such as lexicon and rule-based and Machine Learning (ML) as seen by [4], [8]–[12] but ML has great potential hence this study is focused on ML approach.

The paper’s organization is as follows: Section 2 discusses the related work in SA. Section 3 deals with the crucial stages of the methodology. In Section 4, results are provided and discussed, and finally, the work is concluded.

2. Related Work

Sentiment Analysis (SA) has excellent potential and usage in many domains such as advertising agencies, hospitals, stock exchange, election campaigns, human resources, and supply chain [13]–[16]. Many studies have been conducted to test and improve SA in various domains using different techniques such as; ML approach was used with the Support Vector Machine (SVM) classifier on 1940 reviews. The model was tested using 41 reviews, and the maximum achieved accuracy was 78.05% [8]. Similarly, one more study focused on ML method and used SVM with 79.08% accuracy, Decision Tree (DT) with 75.16%, and Naïve Bayes (NB) with 76.47% accuracy on 400+ tweets [4]. The rule-based approach was used on financial news articles dataset containing 200 rows and achieved 75.6% accuracy [5]. Similarly, one more study used a rule-based technique on

product reviews dataset containing 4,45,509 rows, and achieved accuracy was 72.04% [6]. The lexicon-based approach was used on a dataset containing 6,74,412 rows and achieved 73.5% accuracy [11]. Similarly, one more study experimented with the lexicon approach and achieved 82% accuracy on datasets containing 3,08,316 rows [12]. A study conducted a comparative analysis of 8 classifiers, i.e., SVM, MLP-Deep Learning, K-star, Bayes Net, Simple Logistics, Multi-class Classifier, Decision Tree, and Random Forest in an educational dataset to predict the student's performance. SVM and MLP-Deep Learning were the best performing learning methods and achieved 78.75% and 78.33% accuracy, respectively [17]. Another study compared the Multilayer Perceptron (MLP) and Deep Learning (DL) and achieved 52.60% and 75.03% test accuracy, respectively [18].

Previous studies have mostly used base ML methods in SA but a few studies have also implemented the DL such as [17] and [18] where they only achieved 78.33% and 75.03% accuracies. Not many studies can be found utilizing modern ML methods such as DL and concluding their studies with decent accuracy in SA especially in the domain of opinion mining of people regarding the impact of technology on employment.

This study aims to apply base ML methods along DL to study their performance in SA. The ML methods are applied in a specific domain, i.e., what opinions and sentiments people have regarding the technological advancements and automation taking over their jobs and causing structural unemployment in economies.

This is relatively a new area to explore from this perspective, no study has been conducted in this domain comparing base and modern ML techniques. People having negative sentiments will be afraid of technological advancements and losing their employment to automation. If greater accuracy can be achieved for SA in this domain, that would help to identify the people with negative sentiments so that they can be trained to acquire modern skills to be relevant to the job requirements of the 21st century.

3. Proposed Method

The proposed method has the main four stages

- Data collection
- Data pre-processing
- Sentiment Analysis
- ML Classifiers.

The below sections will discuss each step.

3.1 Data Collection

Ten keywords, also known as "Seed Words," were identified after studying various research articles and the World Economic Forum report that can be used to fetch required text from Twitter [19]. The fetched text was about the technological impact on employment. Hence, seed words like "technology replace human," "technology taking over," and "robots taking over" were used. Each seed word fetched multiple rows of data, and a total of 4,289 rows were fetched between 1st February 2019 to 10th March 2019 in the English language only. The data contained several unrelated and duplicate rows, and the text contained various special characters, symbols, URL's, full stops, periods, colon, hash, quotation marks, braces, brackets, apostrophe, and ellipsis. The next stage will take care of such issues.

3.2 Data Pre-Processing

Various steps under this stage were followed sequentially to reach the final dataset.

- Removal of duplicates unrelated rows, punctuation, rare words, and special characters
- Convert text to lower case
- Tokenization
- Filter stop words and tokens by length
- Stemming
- Vectorization using TF-IDF.

After completing all steps until "Stemming," the dataset was left with 1047 rows. The dataset was broken into N-dimension vector space under the vectorization step to convert it into the integer representation so that it can be solved mathematically by ML classifiers. For this study, the vectorization used the TF-IDF algorithm that works according to the followings [20], [21]:

$$W_{ij} = tf_{ij} \times \log\left(\frac{N}{df_i}\right)$$

where tf_{ij} shows the number of occurrences of i in j , df_i shows the documents that contain i and N shows the total number of the documents.

3.3 Sentiment Analysis

Once the dataset was ready, it was analyzed using a lexical English database known as “WordNet.” The labeling of the rows is an important step that needs to be done before the sentiment can be analyzed. Categorizing all rows into a Positive, Neutral, or Negative polarity is known as labeling process. Many studies have used the WordNet approach for the labeling of text such as [22], [23].

To get the text label by using WordNet, first, the score needs to be calculated for each row in the dataset. WordNet can provide the score either in negative, positive, or zero integer value where a negative value means the row has negative sentiment, and a positive value means the row is positive while zero means it is neutral as shown in Table 1 below

Table 1. Text Labeling Using Wordnet

Sentence	Score	Label
Damn evil robots taking over	-0.25852	Negative
Robots may not be taking over jobs but they are changing them	0	Neutral
Robots taking over to help medical research	0.2	Positive

Table I shows that the first sentence has a negative score hence it has a negative label. Similarly, the score for the second sentence is zero which equals neutral. Finally, the third sentence has a positive score hence a positive label is assigned.

3.4 ML Classifiers

After the labeled data was available, it was the time to train the ML classifiers so that the newer text that belongs to the “technological impact on employment” can be analyzed. For that purpose, the following classifiers were used.

NB: A popular classifier that works on the Bayesian Theorem. It assumes that all features are independent. A good model can be trained using this even if the dataset size is small. It works by assigning the document to its respective class where $P(c|d)$ is maximized, shown below [8].

$$P(c|d) = \frac{P(c) P(d | c)}{P(d)}$$

Where $P(c)$ is class prior probability or belief and $P(d | c)$ is likelihood and $P(d)$ is predictor prior probability.

DT: A classifier that is constructed starting from its root node and continues to go down towards the leaf nodes of the tree. Entropy was used for classification [24] as follows:

$$\sum_{i=1}^c - f_i \log(f_i)$$

where f_i is the frequency of the label i at node and c is the number of unique labels.

SVM: A ML algorithm that can be used to classify the data with respectable accuracy without the need of much fine-tuning. The hyperplane is used for classification according to the following hypothesis function [25]

$$h(x_i) = \begin{cases} +1 & \text{if } w \cdot x + b \geq 0 \\ -1 & \text{if } w \cdot x + b < 0 \end{cases}$$

DL: A modern ML algorithm which is based on Artificial Neural Networks (ANN). DL can be defined as a large ANN that can support big data, and the performance it offers does not slow down when the data is increased. The idea behind DL is, today, more powerful computers are available at affordable cost which can train large ANN or DL within feasible time. The reason it is called “Deep” is that it uses a greedy algorithm that learns through many-layered networks [26,27]. DL was used with “Rectifier” as activation function which is used by the neurons in the hidden layers. A total of 10 epochs was used so that the data can be iterated multiple times. The adaptive rate with 1.0E-8 epsilon and 0.99 rho was used. The data was standardized first with 1.0E-5 L1, 0.0 L2, 10.0 max w2, and an automatic loss and distribution function. To make sure that there is no missing value, the mean imputation missing value handler was used.

The dataset was divided into two parts, i.e., training and validation. 80% of the data was used for training, and 20% was used for validation. A separate file containing 100 rows was used as a test set. Various performance measures can be important to understand the accuracy of ML classifiers such as precision, recall, and accuracy. The accuracy can be calculated by using True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) as shown below.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

The precision can be calculated by using TP and FP, as shown below.

$$\text{Precision} = \frac{TP}{TP+FP}$$

Similarly, the recall can be calculated by using TP and FN, as shown below.

$$\text{Recall} = \frac{TP}{TP+FN}$$

The ML model was set up using Rapid Miner as shown below in Fig. 1. It has training and test set data which is cleaned before going through the classifier.

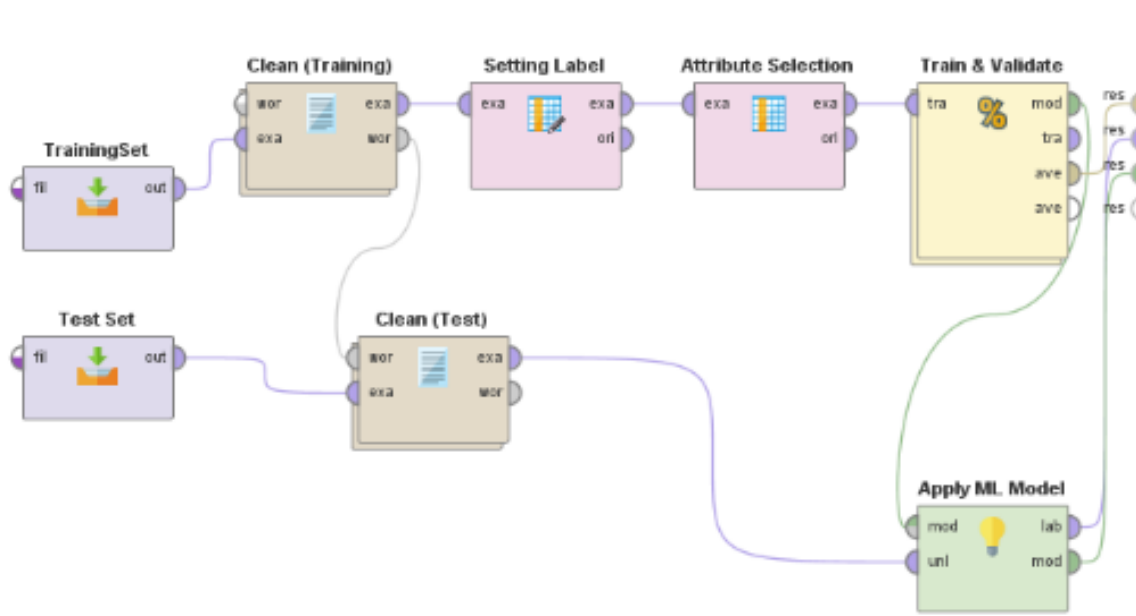


Figure 1. ML Classification Model

4. Results

After setting up the ML classifier model, each algorithm had different accuracy in classifying the test set into {positive, neutral, negative}. The measure to check the performance of each classifier was accuracy, precision, and recall, as shown as (5), (6), and (7). Table II shows the NB classifier’s performance measures.

Table 2. nb classifier performance

Accuracy 87.18%	Performance			
	True Negative	True Neutral	True Positive	Precision
Pred. Negative	83	0	0	100.00%
Pred. Neutral	16	59	2	76.62%
Pred. Positive	7	0	28	80.00%
Recall	78.30%	100.00%	93.33%	

Table II shows that the NB classifier has achieved overall 87.18% accuracy. Next, the performance of the DT will be observed in Table III below.

Table 3. Dt classifier performance

Accuracy 68.21%	Performance			
	True Negative	True Neutral	True Positive	Precision
Pred. Negative	105	40	13	66.46%
Pred. Neutral	0	12	1	92.31%
Pred. Positive	1	7	16	66.67%
Recall	99.06%	20.34%	53.33%	

Table III shows that the DT classifier has achieved overall 68.21% accuracy, which is quite low than achieved by NB. DT classifier suffered the most while classifying the text into a “Neutral” class. Next, the performance of the SVM will be observed in Table IV below.

Table 4. SVM Classifier performance

Accuracy 82.05%	Performance			
	True Negative	True Neutral	True Positive	Precision
Pred. Negative	106	28	2	77.94%
Pred. Neutral	0	26	0	100.00%
Pred. Positive	0	5	28	84.85%
Recall	100.00%	44.07%	93.33%	

Table IV shows that the SVM classifier has achieved overall 82.05% accuracy, which is lower than NB but higher than DT. Next, the performance of the DL will be observed in Table V below.

Table 5. DL Classifier performance

Accuracy 93.33%	Performance			
	True Negative	True Neutral	True Positive	Precision
Pred. Negative	106	13	0	89.08%
Pred. Neutral	0	46	0	100.00%
Pred. Positive	0	0	30	100.00%
Recall	100.00%	77.97%	100.00%	

Table V shows that DL classifier has achieved overall 93.33% accuracy which is highest achieved accuracy as compared to all other algorithms as shown in Fig. 2 below

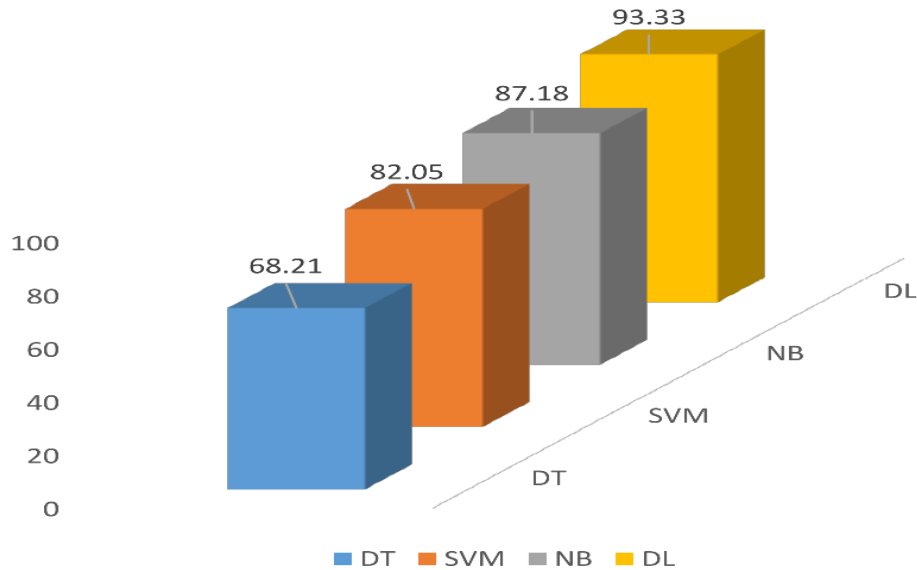


Figure 2. ML Classifiers Accuracy

Fig. 2 depicts the fact that DL has performed better than all other base ML classifiers used in this study. DT has performed the worst while NB was the second-best followed by the SVM in third place.

5. Conclusion

DL performed the best for classifying the text into respective categories. In the future, more data can be collected and provided to DL to test its performance on a massive dataset. There are many parameters under DL which can be adjusted to achieve even better results such as there is a possibility to use many activation functions, i.e., rectifier, tanh, maxout, and exprectifier. More parameters can be explored and experimented to have even better results.

The DL also suffers from some issues of the vanishing or unstable gradients. Each hidden layer becomes significantly slower when it tries to learn. At some points, the performance is degraded up to an extent that it nullifies the benefits of additional layers. Although some modern DL approaches suffer less from such issues but the traditional ANN-based DL approaches are well known for these kinds of issues.

References

1. M. D. Devika, C. Sunitha, and A. Ganesh, "Sentiment Analysis: A Comparative Study on Different Approaches," *Procedia Comput. Sci.*, vol. 87, pp. 44–49, 2016.
2. D. M. E. D. M. Hussein, "A survey on sentiment analysis challenges," *J. King Saud Univ. - Eng. Sci.*, vol. 30, no. 4, pp. 330–338, 2018.
3. S. Sun, C. Luo, and J. Chen, *A review of natural language processing techniques for opinion mining systems*, vol. 36. Elsevier B.V., 2017.
4. V. Vyas and V. Uma, "An Extensive study of Sentiment Analysis tools and Binary Classification of tweets using Rapid Miner," *Procedia Comput. Sci.*, vol. 125, pp. 329–335, 2018.
5. K. Ravi and V. Ravi, *A survey on opinion mining and sentiment analysis: Tasks, approaches and applications*, vol. 89, no. June. Elsevier B.V., 2015.
6. N. Öztürk and S. Ayvaz, "Sentiment analysis on Twitter: A text mining approach to the Syrian refugee crisis," *Telemat. Informatics*, vol. 35, no. 1, pp. 136–147, 2018.
7. F. Colace, M. de Santo, and L. Greco, "Safe: A sentiment analysis framework for e-learning," *Int. J. Emerg. Technol. Learn.*, vol. 9, no. 6, pp. 37–41, 2014.
8. C. Bhadane, H. Dalal, and H. Doshi, "Sentiment analysis: Measuring opinions," *Procedia Comput. Sci.*, vol. 45, no. C, pp. 808–814, 2015.
9. L. I. Tan, W. S. Phang, K. O. Chin, and P. Anthony, "Rule-Based Sentiment Analysis for Financial News," *Proc. - 2015 IEEE Int. Conf. Syst. Man, Cybern. SMC 2015*, pp. 1601–1606, 2016.
10. C.-S. Yang and H.-P. Shih, "A Rule-Based Approach For Effective Sentiment Analysis," *PACIS 2012 Proc.*, 2012.

11. C. Kaushik and A. Mishra, "A Scalable, Lexicon Based Technique for Sentiment Analysis," *Int. J. Found. Comput. Sci. Technol.*, vol. 4, no. 5, pp. 35–56, 2014.
12. M. Z. Asghar, F. M. Kundi, A. Khan, and S. Ahmad, "Lexicon - Based Sentiment Analysis in the Social Web," *J. Basic . Appl . Sci . Res*, vol. 4, no. 6, pp. 238–248, 2014.
13. M. T. Khan and S. Khalid, "Sentiment Analysis for Health Care," *Int. J. Priv. Heal. Inf. Manag.*, vol. 3, no. 2, pp. 78–91, 2015.
14. R. Feldman, "Techniques and applications for sentiment analysis," *Commun. ACM*, vol. 56, no. 4, p. 82, 2013.
15. A. P. Singh, A. Malik, and D. Kapoor, "Sentiment Analysis on Political Tweets," pp. 359–361, 2016.
16. O'Connell Brian, "How Sentiment Analysis and Data Analysis Can Improve Your Sales," *Business News Daily*, 2017. [Online]. Available: <https://www.businessnewsdaily.com/10018-sentiment-analysis-improve-business.html>. [Accessed: 09-Dec-2018].
17. J. Sultana, N. Sultana, K. Yadav, and F. Alfayez, "Prediction of Sentiment Analysis on Educational Data based on Deep Learning Approach," *21st Saudi Comput. Soc. Natl. Comput. Conf. NCC 2018*, pp. 1–5, 2018.
18. A. M. Ramadhani and H. S. Goo, "Twitter sentiment analysis using deep learning methods," *Proc. - 2017 7th Int. Annu. Eng. Semin. Ina. 2017*, 2017.
19. I. Report, *The Future of Jobs Report 2018 Insight Report Centre for the New Economy and Society*. 2018.
20. S. Qaiser and R. Ali, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents," *Int. J. Comput. Appl.*, vol. 181, no. 1, pp. 25–29, 2018.
21. Diggity Matt, "TF*IDF for SEO: The Ultimate Beginner to Advanced Guide," 2019. [Online]. Available: <https://diggitymarketing.com/tfidf-for-seo/>. [Accessed: 29-Mar-2019].
22. A. Srivastava, V. Singh, and G. S. Drall, "Sentiment Analysis of Twitter Data," *Int. J. Healthc. Inf. Syst. Informatics*, vol. 14, no. 2, pp. 1–16, 2019.
23. S. Poria, A. Gelbukh, E. Cambria, P. Yang, A. Hussain, and T. Durrani, "Merging SenticNet and WordNet-Affect emotion lists for sentiment analysis," *Int. Conf. Signal Process. Proceedings, ICSP*, vol. 2, pp. 1251–1255, 2012.
24. S. Ronaghan, "The Mathematics of Decision Trees, Random Forest and Feature Importance in Scikit-learn and Spark," 2018. [Online]. Available: <https://medium.com/@srngn/the-mathematics-of-decision-trees-random-forest-and-feature-importance-in-scikit-learn-and-spark-f2861df67e3>. [Accessed: 18-May-2019].
25. Shuzhan, "Understanding the mathematics behind Support Vector Machines," 2018. [Online]. Available: <https://shuzhanfan.github.io/2018/05/understanding-mathematics-behind-support-vector-machines/>. [Accessed: 18-May-2019].
26. Mudavath, M., Kishore, K.H., Hussain, A., Boopathi, C.S. (2020). Design and analysis of CMOS RF receiver front-end of LNA for wireless applications. *Microprocessors and Microsystems*, 75, art. no. 102999.
27. J. Brownlee, "What is Deep Learning?," 2016. [Online]. Available: <https://machinelearningmastery.com/what-is-deep-learning/>.