**How to Cite**:
Monika, P., Kulkarni, C., Harish Kumar, N., Shruthi, S., & Vani, V. (2022). Machine learning approaches for sentiment analysis: A survey. *International Journal of Health Sciences*, *6*(S4), 1286–1300. https://doi.org/10.53730/ijhs.v6nS4.6119

# Machine learning approaches for sentiment analysis: A survey

**Monika P**
BMS College of Engineering, Bangalore, India

**Chaitanya Kulkarni**
Dayanada Sagar College of Engineering, Bangalore, India

**Harish Kumar N**
Dayanada Sagar College of Engineering, Bangalore, India

**Shruthi S**
Dayanada Sagar College of Engineering, Bangalore, India

**Vani V**
Nitte Meenakshi Institute of technology, Bangalore, India

***Abstract*---**Sentiment Analysis or Opinion Mining is popular task of Natural Language Processing (NLP) performed on textual data generated by users to know the orientation or sentiment of the text. To perform Sentiment Analysis, it is critical to create an accurate and precise model, machine learning techniques are heavily utilized to build an accurate model. Deep learning and transfer learning techniques have been found to have increased utilization and better results, making them one of the most popular research areas around the world. Hotel and restaurant industries analyze reviews to obtain a deeper understanding of their client's needs, likes and dislikes, whereas specialists use Twitter data and stock market news items to forecast stock market trends. Machine Learning algorithms are most essential part of a Sentiment Analysis model, this survey paper analyze all the widely used Machine Learning Approaches for Sentiment Analysis. A brief introduction on Methodology for Sentiment Analysis is given along with conclusion and future scope and in the field of Sentiment Analysis.

***Keywords*---**sentiment analysis, natural language processing, machine learning, artificial intelligence.

International Journal of Health Sciences ISSN 2550-6978 E-ISSN 2550-696X © 2022.
***Corresponding author**: Monika P; Email: monika.mel@bmsce.ac.in*
*Manuscript submitted: 27 Jan 2022, Manuscript revised: 18 Feb 2022, Accepted for publication: 9 March 2022*

## Introduction

Social media and web browsing is booming across the globe as result of which the user generated data is also increasing day by day at a very high rate. It is estimated that 175 Zettabytes of data will be accumulated by the year 2025. The introduction of IOT is also playing an important role in generation of real time data in various forms. The data is being collected and monitored in various forms such as images, texts, videos, audios etc. One of the important and useful means of data being Text data is being processed to produce useful outcomes. Social media and webs are constantly producing huge amount of text data as people are very open in sharing views, ideas and opinions in such platforms.

Natural Language Processing is one of the emerging and booming fields of Artificial Intelligence (AI) which deals with text data. NLP is regarded as the intersection of Computer Science (CS), Machine Learning (ML) and Linguistics. NLP is the ultimate guide to make computers understand language as through and deeply as humans do. Although NLP has managed to produce a lot of breakthroughs in the field of AI it is not yet computationally cheap and fast as it involves training of Algorithms and Neural networks like Recurrent Neural Network (RNN), (Hochreiter and Schmidhuber, J 1997) LSTM, Bi-LSTM etc. After the introduction of (Vaswani et al.,2017) Transformer's network and BERT (Devlin et al, 2018) as the time required for training classifier reduced as they used Transfer learning technologies hence only need tweaking rather than training It from scratch. The main challenge of Natural Language processing or text data is handling the different linguistic rules for all the languages as the rules and styles differ from each language therefore, a model trained on English dataset cannot be used for a Hindi dataset or any other language. Another major challenge is the informal and vague way of writing followed by individuals, people tend to make a lot of grammatical errors, spelling errors etc. There is a lot of research going on in this subject to make it a simple and practical solution for texts.

Sentiment Analysis (SA) or Opinion Mining (OM) is one of the emerging and important application of NLP and ML used widely for lot of applications. Sentiment Analysis is a process of analysing the sentiments or orientations of the user generated text or reviews. This can be achieved using a lot of approaches like statistical approach, dictionary approach, semantic approach and Machine Learning approach. Lexicon based approaches like dictionary based, semantic approach and rule-based approaches makes use of knowledge of languages and statistics to predict the sentiment of the text but has a lot of disadvantages as the model will be domain specific and cannot be utilized for texts of other domains as the sentiment of the word vary domain to domain. For example, 'BIG' may be positive in a review in electronics domain whereas same token 'BIG' may be negative in travel domain. Machine Learning approach can overcome these dis-advantages and give better results when trained with lots of data and hyperparameters tweaked with precision. Lot of algorithms like Support Vector Machines, Naïve Bayes, Maximum Entropy, Decision Trees are being used in this approach to provide better results. Now-a-days Deep learning has shown promising results in Sentiment Analysis and Aspect detection. Researchers have proposed lot of models using ANNs, RNNs, LSTMs, CNNs, Bi-LSTM, Transformers networks. The only dis-advantage of deep learning technique is its long training time as the model is very complex and requires a lot of computational cost and is slow in training.

In this survey paper various Machine Learning and deep learning algorithms used in Sentiment analysis is discussed. We have tabulated the results of various approaches used by researchers.

**Methodology**

The general methodology for sentiment analysis is shown in Figure 1. There are mainly 5 significant steps involved in Sentiment Analysis.

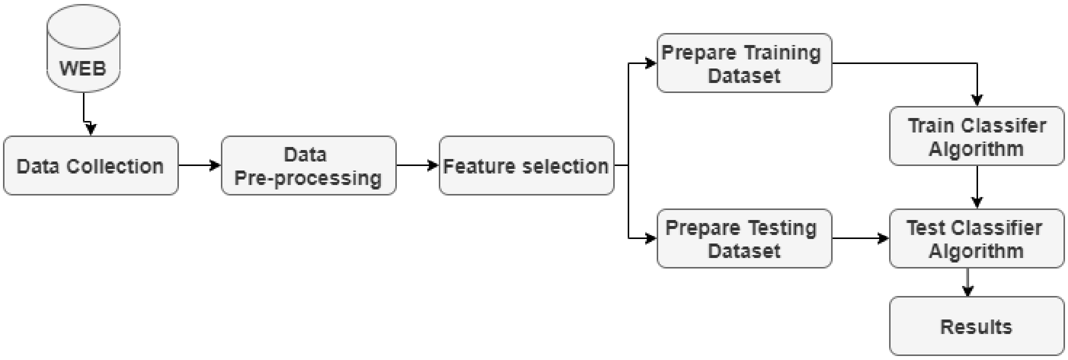


*Figure 1 General Methodology for Sentiment Analysis*

**Data Collection**

There are numerous ways for data collection. There are pre-collected datasets available in many websites like Kaggle, UCI Machine Learning Repository etc. There are lots of variety of data set available for all NLP tasks covering lot of domains like healthcare, entertainment, travel and tourism etc. Twitter is also a popular means for data collection. Tweets from various domains can be collected from twitter by tagging appropriate key words and hashtags. Tweepy is a popular library used to collect tweets from twitter. Realtime tweets can also be collected using Tweepy libraries Stream function. Data may be collected from various websites and blogs using APIs and libraries.

**Data Pre-processing**

This one of the key steps of any NLP task (Kannan et al., 2014), the raw text data collected in previous step has to be pre-processed in order to clean the data and optimize the classification (Etaiwi, W. and Naymat 2017). There are number of pre-processes which needs to be done to the raw text to feed in to next step. Figure 2 shows few key steps of NLP pre-processing

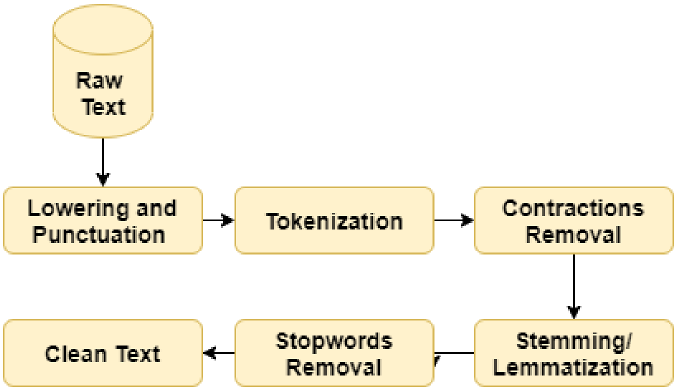


*Figure 2 General Flow diagram for Pre-processing*

A] Lowering and Punctuation- This is the basic step of NLP pre-processing where the text is lowered to avoid duplication and punctuations are removed to clean the text. Punctuation adds unnecessary information to the data which should be removed to decrease computational cost and get better results. For example, consider a text "It is Good……." Will be converted to "it is good" the length of the text is reduced but the meaning remains the same and the orientation of the text is constant. By lowering both words "Good" and "good" will be same avoid storing 'good' multiple times.

B] Tokenization- This is the fundamental step of pre-processing where we divide the text into smaller chunks or tokens to make further process easy.Tokens are the minimum units, usually words. These tokens are further converted to vectors or numerical value to feed to a classifier. For example, a review r1=" the food was good" after tokenization will be r1= ['the', 'food', 'was', 'good'].

C] Contraction's Removal- In this step we remove contractions present in the raw text. Contractions refer to shortcuts used in the language. For example, is not is contracted as isn't, can not is contracted as can't.

D] Stemming- Stemming is a process of conversion of a token into its stem form or root form. The prefixes and suffixes are removed in this step as they don't add any significant meaning to the token. For example, go, going, gone will be converted to go as the base meaning and orientation of all the words Is the same. This will help to reduce the size of the vocabulary

E] Lemmatization- Lemmatization is the process of conversion of tokens to their lemmas. Morphological variations of same word is avoided by converting tokens to their base form or lemmas.

F] Stopwords removal- This is the most important step in NLP pre-processing which involves the removal of unnecessary words which don't add any meaning to the sentence and will not contribute in deciding the sentiment or orientation of the sentence. Words like the, is, for, am, I etc are some examples of Stopwords, there are predefined set of Stopwords available in each language. Removing Stopwords will help reduce the vocabulary size without affecting the performance of the model.The raw text when passed through the above pipeline of NLP pre-processing will be cleaned with all unimportant token being removed and sentences converted to set of tokens ready for next step.

**Feature Extraction**

Feature extraction is a process of extracting important information from text data. It is process of conversion of text data into numbers or vectors. There are number of ways to convert text to number starting from Bag of words which is the simplest to latest advancements like Glove and fast text.

A] Bag of words- This is one of the simplest approaches of feature extraction or text modelling used in NLP. BOW is process where we keep a track of occurrence of each word in the text or document (Zhang et al., 2010). Each word or token is regarded as a feature in BOW model and the syntactic and semantic meaning of the text is completely ignored. The 'bag' represents the vocabulary made of unique words from the training dataset. For instance, consider two sentences s1=" the view was bad" and s2=" the food was tasty". To covert these into vector using the Bag of Words method, the vocabulary is created using both the sentences. V= { 'the', view' , 'was', 'bad', 'food', 'tasty'} and s1 is represented as s1=[1 1 1 1 0 0] and s2=[1 0 1 0 1 1]. Table 1 represents the tabulated form of sentences.

Table 1 Bag of Words Representation

| Tokens | the | view | was | bad | food | Tasty |
|---|---|---|---|---|---|---|
| S1 | 1 | 1 | 1 | 1 | 0 | 0 |
| S2 | 1 | 0 | 1 | 0 | 1 | 1 |

B] TF-IDF- TF-IDF or Term Frequency – Inverse Document Frequency is a feature extraction method in NLP which also reflects how important a token Is in the document in corpus of documents (Ramos et al., 2003).BOW model weights all the words equally whereas the TF-IDF model weights words according to their occurrence in each sentence. The two basic formulas used for TF-IDF are given by:

*Term Frequency (TF)* – Number of times token t appears in a sentence / Total Number of tokens in a sentence (1)

*Inverse Document Frequency (IDF)* – Log (Total number of sentences / Number of sentences with token t in it) (2)

For instance, consider two sentences s1=" the view was bad" and s2=" the food was tasty". To covert these into vector using TF-IDF method, the vocabulary is created using both the sentences. V= {'the', view', 'was', 'bad', 'food', 'tasty'}.

*Table 2 Term Frequencies*

| Tokens | the | view | was | bad | food | Tasty |
|---|---|---|---|---|---|---|
| S1 | ½ | ½ | ½ | ½ | 0 | 0 |
| S2 | ½ | 0 | ½ | 0 | ½ | ½ |

*Table 3 Inverse Document Frequencies*

| Tokens | the | view | was | bad | food | Tasty |
|---|---|---|---|---|---|---|
| S1 | 0 | 0.3465 | 0 | 0.3465 | 0 | 0 |
| S2 | 0 | 0 | 0 | 0 | 0.3465 | 0.3465 |

C] Word2vec- The vectorisation of tokens is done using a 2-layer neural network called word2vec. It is Mikolvo's Le, Q. and Mikolov, T., (2014) most well-known and widely utilised vectorizing techniques. Continuous bag of words (CBOW) and skip gram model are the two main models in Word2vec. The Continuous Bag of Words (CBOW) model predicts the target word using context words, whereas the Skip Gram model predicts the Context words using the target word. Window size may be passed as a hyper-parameter to choose the context size. For a larger dataset, (McCormick et al.,2016) the Skip Gram model offers superior results.

Training and Testing We need to spilt the data for training and testing in order to train the classifier using the training dataset and test the model using the testing dataset which we separated from the original dataset. (Arlot and Celisse, 2010) K-fold cross validation may also be performed on the dataset where we train model with k-1 folds and test it with 1-fold.

**Results and discussion**

The final step is to publish or tabulate the results. There are multiple metrics used to publish results like accuracy, precision, Recall, F-score, Sensitivity, Specificity. The most commonly used metric is accuracy while the metric has to be choose based on the domain. Confusion Matrix Is another popular means to show the results which tabulates all classifications and mis-classifications done by the model. Figure 3 shows a confusion matrix (Townsend ,1971). Where True Positives represents samples that are actually true and classified as true, True Negatives represents samples that are actually false and classified as false, False Negatives are the samples that are actually true but are mis-classified as false whereas False positives are the samples that are actually false but mis-classified as True. All the metrics mentioned above can be calculate with the help of Confusion Matrix.

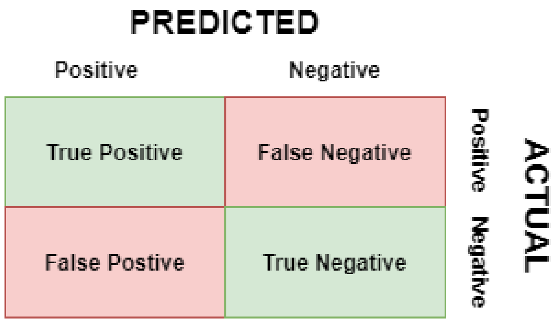


*Figure 3 Confusion Matrix*

**Machine Learning Approaches**

Machine Learning (ML) Algorithms can be used to categories sentiments (Agarwal B and Mittal N, 2016). This assignment can be accomplished using both supervised and unsupervised learning methods. Because of its accuracy, supervised learning methods are increasingly often used. Before being applied to real data, these algorithms must be trained on a training set. Figure 4 has the flowchart of various Machine Learning Approaches used for Sentiment Analysis.

**Un-Supervised Learning**

Un-supervised Learning is a popular type of Machine Learning approach which does not require any labelled data to train on. Collecting Labelled data or labelling huge corpora is a tedious and laboring task. Therefore, unsupervised learning methods may be used for such times as collection of un-labelled data is an easy task. The statistical properties such as word co-occurrence, lexicons with polarized words are used for clustering which can classify data into categories without specifying the type of it (Tripathy et al.,2016).

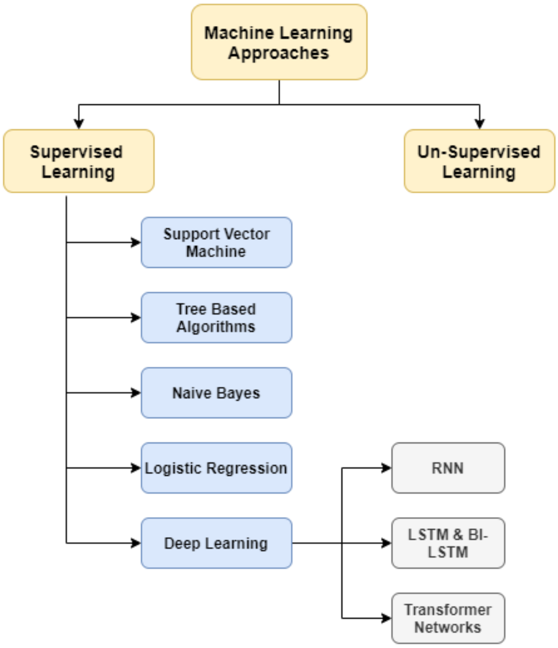


*Figure 4 Machine Learning Approaches for Sentiment Analysis*

**Supervised Learning**

Supervised Learning is a type of Machine Learning approach where the model is trained using the labelled data. The algorithm constantly learns using the training data tweaking and altering the weights. In NLP task Supervised Learning Algorithms are used more oftenly because of its high accuracy (Tripathy et al., 2016). The most popular supervised learning used for sentiment analysis are:

**Support Vector Machine Support**

Vector Machine (SVM) is one of the most popular supervised learning algorithms implemented for sentiment analysis (Moraes et al., 2013). SVM's have shown great accuracies in many NLP tasks including Sentiment Analysis. The primary task of SVM is to create the best hyperplane or boundary that best divides the data into classes so that we can place the unknown values in future and categorize it with at most precision. SVM selects the extreme points or vectors that aid in the formation of the hyperplane. Support vectors are the extreme instances, and the algorithm is called a Support Vector Machine. Consider the example shown in Figure 5 which shows maximum margin hyperplane which best separates class 2 and 1

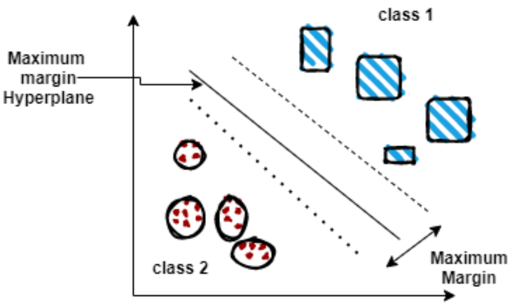


*Figure 5 Support Vector Machine Classifier*

Table 4 has tabulated results of various proposed methods using Support Vector Machine along with their results. SVM has shown some remarkable performance in Sentiment Analysis giving accuracies as high as 90 percent.

| No. | Dataset | Algorithm | Dataset size | Result |
|---|---|---|---|---|
| **[18]** | Movie, book, product review | SVM-Unigram | 2000 each domain | 84.2% Fscore |
| **[19]** | Twitter tweets | SVM (c=0.01) | 1000 tweets | 89.365% Accuracy |
| **[20]** | SemEval 2015 Task 11 | Linear SVM | 110,000 tweets | 83.4% Accuracy |
| **[21]** | Turkish Twitter data | RF+SVM | 3000 and 10500 | 86.4% Accuracy |
| **[22]** | NLPCC2014 emotion analysis evaluation task | CNN-SVM | 10,000 | 88.8% Accuracy |

Table 4

**Naïve Bayes**

Naïve bayes is a probabilistic classifier which uses Bayes theorem for solving classification problem (Murphy et al ,.2006)The conditional probability of occurrence of X given the individual probalities of X and Y and conditional probability of occurrence of event Y. It is presumed that features are independent in this case. Naïve bayes is preferred when there is a small training data available.

$$P(X/Y) = P(X)*P(Y/X) \,/\, P(Y) \qquad (3)$$

For finding the sentiment of the text the equation may be rewritten as:

$$P(sentiment/text) = P(sentiment) * P(text/sentiment) \,/\, P(text) \qquad (4)$$

Where,

P (sentiment) = probability of sentiment
P(text/sentiment) = probability of text classified as a sentiment
P(text) = probability of occurrence of text

**Logistic Regression**

Logistic Regression is a type of regression analysis used for Classification tasks. Logistic Regression is widely used for binary classification tasks (Wright et al.,1995). Logistic regression gets the ratio of odds when there are more than single explanatory variables. The key benefit is that it eliminates confusing effects by examining the relationship between all variables. The probability of an outcome by logistic regression is given by:

p(Yi) = 1 / (1+ e^(-b0+b1X1i)
(5)

where,
p(Yi) = probability that Y is 1 or True for case i e= constant (2. 72..)
b0 = Constant (estimated from data)
b1= b-coefficient (estimated from data)
Xi = input or observed value of X for case i

**Tree Based Algorithms**

Tree based algorithms are a common supervised learning approach where the target variable is decided by building a tree like structure. Tree based algorithm use gini impurity or information gain to decide the root node which best divides the data and further nodes are calculated using the same method. There are many algorithms available to build a decision tree such as ID3 (Jin et al.,2009) CART, C4.5, C5.0 (Hssina et al, 2014) etc. Decision Tree algorithm is seen to suffer with high variance even though it has a low bias. Random Forest is a popular alternative which uses bagging technique and uses multiple trees and soft votes or hard votes to aggregate all the decisions and arrive at a final decision. In recent times, boosting techniques like adaboost (Hastie et al., 2009), gradient boost (Ke et al., 2017) and XgBoost (Chen et al.,2016) have shown good accuracies in lot of classification tasks.

**RNN**

Recurrent Neural Networks popularly known as RNN's is an interesting type of Neural Network used for text data or sequential data (Mikolov et al., 2010). Recurrent Neural Networks are commonly used in NLP, image captioning, language translation, speech recognition etc. In vanilla Neural Network the inputs and outputs are independent of each other, but in RNN's the current step is dependent on the previous step as the output from previous step is fed as input step in current step. As the RNN's uses previous prior time stamp information is used to forecast the current time stamp, ensuring that previous information is used and acting as memory by remembering certain sequence information. The fundamental drawback of a typical Recurrent Neural Network is that it suffers from vanishing and exploding gradient descent (Zhou et al., 2019), making it unable to remember very long-term dependencies in the sequence. Recurrent Neural Network has been widely used in various NLP tasks including Sentiment Analysis and have proven to be great choice by giving good accuracies when trained on enough data.

| No. | Dataset | Algorithm | Dataset size | Result |
|---|---|---|---|---|
| [38] | Product reviews | Tree bank and fuzzy opinion mining model, naïve bayes | 2000 product reviews | 58.58 Accuracy |
| [39] | Twitter | Naïve bayes multi label | 39,095 tweets of Purdue university | 55.19 Accuracy |
| [40] | Twitter (KFC and McDonalds related) | Maximum Entropy | 14,000 Tweets | 78% validation Accuracy |
| [41] | StockTwits data for 5 companies | logistic regression - TFIDF | 175,000 # of tweets with sentiment | Avg 60% Accuracy |
| [42] | BBC news dataset | lexicon based approach | 2225 articles | NA |
| [43] | Facebook posts | K means Clustering | 1000 | 87% Recall |

Table 5

**LSTM and BI-LSTM**

Long Short-Term Memory or LSTM originated as a replacement for an RNN that did not have to deal with vanishing and exploding gradients like RNNs do. LSTMs have gates that tell them what information to remember, what information to send to the next time step, and what memory to forget from the past. LSTM's have 4 gates cell state, input gate, output gate along with a forget gate. They are one of the most widely used models due to their capacity to forget and retain needed information. In most NLP applications, an RNN cell is substituted with an LSTM cell. The biggest disadvantage of utilising an LSTM is that it takes a long time to train and consumes a lot of processing resources. In addition, the input must be passed consecutively. Bi-LSTMs have recently been used in practically all NLP (Zhou et al., 2019), use cases since they not only fix the shortcomings of RNNs, but they also employ past and next time step information to anticipate the current time step, as we transmit the sequence in both forward and backward directions. LSTM and BI-LSTM are being used as substitution for RNN from a long time as there are lot of advantages of them over an RNN. Sentiment Classification (Shoryu et al,.2021; Muhammad et al,2021) and Aspect identification (Gandhi and Attar, 2020; Kumar and Abirami S, 2021). are some popular tasks LSTM and BI-LSTMs are used for.

## Transformer Networks

In the year 2017, Google Brain researchers, in collaboration with Google Research and the University of Toronto, published "Attention is all you need," a paper that transformed NLP applications. This model is a stack of encoder and decoders that include self-attention layer, multi-headed attention layer, normalizations, and feed forward layers. The inputs are word embeddings and a position vector that describes the input sequences positions, and unlike other models that accept serial inputs, the inputs can be delivered in parallel. Self-attention is calculated for each token in the encoder component using a query vector and key-value pair. This is done numerous times and piled on top of each other to generate a multiheaded attention layer that is passed to the feed forward layer. The model has six encoder layers and six decoder layers. The two vectors Key and Value are sent from the encoder to the decoder. The decoder layer is made up of three layers: the first is the self - attention layer, which is followed by a normalization layer that is identical to the encoder layer, and the second is the encoder decoder attention layer. A feed forward network with a linear and a SoftMax layer is used to construct an output vector using the output self-attention and the input from the encoder. Sequential output is produced by a transformer network, for improved outcomes, there are few skip connections or residuals present in both encoder part and decoder part. The key benefit is that because Transformers do not use recurrence, they do not suffer from vanishing or exploding gradient problems, and also, they are faster and inexpensive to train. Transformers and attention networks are widely being used in Sentiment Analysis (Naseem et al,.2020) to map out sentiment words and to find Aspects (Javdan, S. and Minaei-Bidgoli,2020) in Aspect Based Sentiment Analysis which is one of the emerging and most widely used variant of Sentiment Analysis. Pre-trained networks are already available for transfer learning which ensures fast training and good accuracy (Zhang et al., 2020). Table 5 has tabulated results of various methods proposed by researchers for sentiment analysis using various Machine Learning algorithms. The results are as high as 99 percent when Random Forest and Bagging technique was used on twitter tweets.

## Conclusions

Sentiment Analysis being one of the hot topics for researchers across the globe has caught attention of AI/ML/NLP researchers. Lots of Machine Learning (ML) and Deep Learning (DL) algorithms are being applied on text data to extract useful information. Sentiment Analysis is a process of extracting of Opinion and orientation of the author or user through his review or text data. There is wide variety of Algorithms available in ML that can be applied for SA. All the algorithms are discussed and results of previous works have been tabulated. Survey paper has also discussed various approaches for Sentiment Analysis using various Deep Learning approaches. In this survey paper, we tried using Flowcharts and tables for cogent understanding of all approaches. Future scope includes using other approaches like Transfer learning and hybrid models to build more complex models for complicated tasks like aspect extraction and sentiment classification.

Using Pre-trained models like BERT and GPT's also have a good scope in tweaking models to perform sentiment analysis.

**References**

Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. Hochreiter, S. and Schmidhuber, J., (1997). Long short-term memory. Neural computation, 9(8), pp.1735-1780.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998- 6008).

Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J., Nithya, M., Kannan, S. and Gurusamy, V., (2014). Preprocessing techniques for text mining. International Journal of Computer Science & Communication Networks, 5(1), pp.7-16.

Etaiwi, W. and Naymat, G., (2017). The impact of applying different preprocessing steps on review spam detection. Procedia computer science, 113, pp.273-279.

Zhang, Y., Jin, R. and Zhou, Z.H., (2010). Understanding bag-of-words model: a statistical framework. International Journal of Machine Learning and Cybernetics, 1(1-4), pp.43-52.

Ramos, J., (2003), December. Using tf-idf to determine word relevance in document queries. In Proceedings of the first instructional conference on machine learning (Vol. 242, No. 1, pp. 29-48).

Le, Q. and Mikolov, T., (2014 June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196). PMLR. McCormick, C., (2016.) Word2vec tutorial-the skip-gram model. Apr-2016.[Online]. Available: http://mccormickml. com/2016/04/19/word2vec-tutorial the-skip-gram-model.

Jang, B., Kim, I. and Kim, J.W., (2019). Word2vec convolutional neural networks for classification of news articles and tweets. PloS one, 14(8), p.e0220976.

Arlot, S. and Celisse, A., (2010). A survey of cross-validation procedures for model selection. Statistics surveys, 4, pp.40-79.

Townsend, J.T., (1971). Theoretical analysis of an alphabetic confusion matrix. Perception & Psychophysics, 9(1), pp.40-50.

Agarwal, B. and Mittal, N., (2016). Machine learning approach for sentiment analysis. In Prominent feature extraction for sentiment analysis (pp. 21-45). Springer, Cham.

Tripathy, A., Agrawal, A. and Rath, S.K., (2016). Classification of sentiment reviews using n-gram machine learning approach. Expert Systems with Applications, 57, pp.117-126.

Moraes, R., Valiati, J.F. and Neto, W.P.G., (2013) .Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 40(2), pp.621-633.

Murphy, K.P., (2006). Naive bayes classifiers. University of British Columbia, 18(60), pp.1-8. Wright, R.E., (1995). Logistic regression.

Agarwal, B., Poria, S., Mittal, N., Gelbukh, A. and Hussain, A., (2015). Concept-level sentiment analysis with dependency-based semantic parsing: a novel approach. Cognitive Computation, 7(4), pp.487-499.

Ramasamy, L.K., Kadry, S., Nam, Y. and Meqdad, M.N., (2021). Performance analysis of sentiments in Twitter dataset using SVM models. International Journal of Electrical & Computer Engineering (2088-8708), 11(3).

Karanasou, M., Ampla, A., Doulkeridis, C. and Halkidi, M., (2016, December). Scalable and real-time sentiment analysis of twitter data. In 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW) (pp. 944-951). IEEE.

Shehu, H.A. and Tokat, S., (2019, April). A hybrid approach for the sentiment analysis of Turkish Twitter data. In The International Conference on Artificial Intelligence and Applied Mathematics in Engineering (pp. 182-190). Springer, Cham.

Chen, Y. and Zhang, Z., (2018, May). Research on text sentiment analysis based on CNNs and SVM. In 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA) (pp. 2731-2734). IEEE.

Jin, C., De-Lin, L. and Fen-Xiang, M.,( 2009, July). An improved ID3 decision tree algorithm. In 2009 4th International Conference on Computer Science & Education (pp. 127-130). IEEE.

Hssina, B., Merbouha, A., Ezzikouri, H. and Erritali, M., (2014). A comparative study of decision tree ID3 and C4. 5. International Journal of Advanced Computer Science and Applications, 4(2), pp.13-19.

Hastie, T., Rosset, S., Zhu, J. and Zou, H., (2009). Multi-class adaboost. Statistics and its Interface, 2(3), pp.349-360.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y., (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30, pp.3146- 3154.

Chen, T. and Guestrin, C., (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

Mikolov, T., Karafiát, M., Burget, L., Černocký, J. and Khudanpur, S., (2010). Recurrent neural network based language model. In Eleventh annual conference of the international speech communication association.

Pascanu, R., Mikolov, T. and Bengio, Y., (2013, May). On the difficulty of training recurrent neural networks. In International conference on machine learning (pp. 1310-1318). PMLR.

Zhou, J., Lu, Y., Dai, H.N., Wang, H. and Xiao, H., (2019). Sentiment analysis of Chinese microblog based on stacked bidirectional LSTM. IEEE Access, 7, pp.38856-38866.

Gandhi, H. and Attar, V., (2020). Extracting aspect terms using CRF and biLSTM models. Procedia Computer Science, 167, pp.2486-2495.

Kumar, J.A. and Abirami, S., (2021). Ensemble application of bidirectional LSTM and GRU for aspect category detection with imbalanced data. Neural Computing and Applications, pp.1-19. Shoryu, T., Wang, L. and Ma, R., (2021, May). A Deep Neural Network Approach using Convolutional Network and Long Short-Term Memory for Text Sentiment Classification. In 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD) (pp. 763- 768). IEEE.

Muhammad, P.F., Kusumaningrum, R. and Wibowo, A., (2021). Sentiment analysis using Word2vec and long short-term memory (LSTM) for Indonesian hotel reviews. Procedia Computer Science, 179, pp.728- 735.

Naseem, U., Razzak, I., Musial, K. and Imran, M., (2020). Transformer based deep intelligent contextual embedding for twitter sentiment analysis. Future Generation Computer Systems, 113, pp.58-69.

Javdan, S. and Minaei-Bidgoli, B., (2020, July). Applying transformers and aspect-based sentiment analysis approaches on sarcasm detection. In Proceedings of the Second Workshop on Figurative Language Processing (pp. 67-71).

Zhang, T., Xu, B., Thung, F., Haryono, S.A., Lo, D. and Jiang, L., (2020, September). Sentiment analysis for software engineering: How far can pre-trained transformer models go?. In 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 70-80). IEEE. Chauhan, C. and Sehgal, S., (2017, May). Sentiment analysis on product reviews. In 2017 International Conference on Computing, Communication and Automation (ICCCA) (pp. 26-31). IEEE.

Chen, X., Vorvoreanu, M. and Madhavan, K., (2014). Mining social media data for understanding students' learning experiences. IEEE Transactions on learning technologies, 7(3), pp.246- 259

El Rahman, S.A., AlOtaibi, F.A. and AlShehri, W.A., (2019, April). Sentiment analysis of twitter data. In 2019 International Conference on Computer and Information Sciences (ICCIS) (pp. 1-4). IEEE.

Gupta, R. and Chen, M., (2020, August). Sentiment Analysis for Stock Price Prediction. In 2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 213-218). IEEE.

Taj, S., Shaikh, B.B. and Meghji, A.F., (2019, January). Sentiment analysis of news articles: A lexicon based approach. In 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-5). IEEE.

Neri, F., Aliprandi, C., Capeci, F., Cuadros, M. and By, T., (2012, August). Sentiment analysis on social media. In 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (pp. 919-926). IEEE.