# A STUDY ON TWITTER SENTIMENT ANALYSIS USING DEEP LEARNING

**Sathya M[1], Thulasimani K[2], DevaSurithi S[3]**

*[1,3] P. G. Student, Department of Computer Engineering, GCE, Tirunelveli, TamilNadu, India*
*[2] Professor, Department of Computer Engineering, GCE, Tirunelveli, TamilNadu, India*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract -** *Sentiment analysis is a branch of research that examines feelings, attitudes, and reviews from many public spheres. Now-a-days, people share their thoughts and insights on a wide range of issues and topics via social media. Recently social networking sites like Twitter and Face book have become popular because users can able to express their opinions with other internet users through micro blogging. Today Twitter is among the most widely used blogging sites. But the disrespectful, insensitive, or unfair remarks that occasionally appear in online forums drive many people away. The majority of websites are unable to promote productive discourse, thus either heavily restrict or fully disable user comments. Insightful data about what is stated on Twitter is provided when sentiment analysis is combined with Twitter. This study analyzed with various deep-learning techniques for the classification of negative and positive elements. Data set SemEval-2017 from Twitter is used to train the final models and will be useful to identify the model which produces the most accurate results.*

***Key Words***: **Sentiment analysis, social media, Twitter, tweets, positive, negative, neutral, deep learning**

## 1. INTRODUCTION

Twitter had developed to turn into a wellspring of fluctuated sort of data because of the nature of small-scale writes on which individuals post continuous messages about their suppositions on an assortment of themes, examines current issues, whine, and express positive assessment for items used in day-by-day life. The primary goal of this study is to conduct sentiment analysis on tweets utilizing different deep learning algorithms that classifies the tweets into the positive or negative category. If a tweet contains both positive and negative elements, the final message should be chosen based on which element is more prominent. Emojis, usernames, and hashtags in the tweets must be analyzed and converted into a standard format.

However analyzing the sentiment expressed is not an easy task. There are several problems in terms of tonality, polarity, lexicon, and tweet grammar. It seems to be highly informal and pseudo-grammatical and it's going to be hard to grasp their background. In contrast, the regular use of slang words, acronyms, and vocabulary words are very popular when posted online. The categorization of such terms by polarity is difficult for the natural processors involved. The identification of negative, neutral, and positive tweets are obtained using Bidirectional- Attention-based LSTMs, CNNs, and fine-tuning Google's pre-trained BERT architecture, which has generally performed as a state of art for most NLP tasks.

## 2. RELATED WORK

Numerous studies have been conducted on fully automated systems that extract features from datasets devoid of human involvement [2] Utilizing novel features like DAL scores and n-grams, among others, the sentiment analysis for categorization was done at the phrasal level. Syntactic details' polarity [4, 5] was employed as a feature. However, this approach need a precise expression border to capture the intended mood. Due to the difference in how words are produced using DAL, which is not a component of speech, it also cannot manage polysemy.

VADER is a straightforward rule-based model for broad sentiment analysis[6], and contrasted its performance with well-known state-of-the-art benchmarks like SentiWordNet, LIWC, ANEW, and machine learning methods like Naive Bayes and Support Vector Machine (SVM) algorithms. Express and emphasize sentiment, VADER blends the lexical elements with general grammatical and syntactical norms.

Analyzing the tweets in written English that come from various KSA telecommunications firms[7] and used supervised machine learning techniques for classification to execute opinion mining on them. Gauging the tweet, how essential a word is used in a particular tweet, and also employed with TF-IDF (Term Frequency-Inverse Document Frequency).

Sentiment analysis approaches embedded in public Arabic tweets and Face book comments[8]. Supervised machine learning algorithms such as Support Vector Machine (SVM) and Naïve Bayes, are Binary Model(BM) and TF-IDF returns the effect of several terms weighting functions on the accuracy of sentiment analysis. Using natural language analysis for Arabic language text[9] sentiment analysis is applied on the Twitter dataset of 4700 for Saudi dialect sentiment analysis with (k=0.807).

Egyptian dialect using a corpus such as tweets, and products review data for sentiment analysis[10]. Natural language processing technique is applied to understand the Egyptian dialect and additionally, used a lexicon-based classification method to classify the data. Using Support Vector Machine for emotion analysis[11], collected optimistic, negative, and neutral tweets from several sources, including the Sentiment140 database. The features are extracted from each message, such as characters n-grams, number of hash-tags, emoticons, etc., for classification and the F1 score of 69.02 is obtained in three-way classification. Similar studies in the assessment of document classification techniques in [12], Bi-LSTM model returns the strongest result for tweet three-way classification and F-score value is 68.5 using SemEval dataset.

In the course of a few years, sentimental computing has set foot in the area of machine learning as social media have been used by different abusers. Also, these are required abilities for many Human-computer interaction applications. Lot of studies have been done on sentimental and content analysis in a combined manner for identifying and interpreting human emotional messages[13]. Apart from the traditional parsers that were based on normal searching, a greedy parsing technique named Transition based dependency is used for the classification. This parser will give more accuracy of the classification results

## 3. SYSTEM ARCHITECTURE

Pre-processing of training data uses natural language processing which is a technique accustomed to perceiving computer information and handling human interactions. The text comments that are given to the model are additionally being pre-processed. Both pieces of information are passed to the sentiment library where the feature extraction of the pre-processed information is being done. From this, we tend to get the trained model for the knowledge sets. The system architecture is depicted in Fig.1. The classification is finished using LSTM, which is another version of the Recurrent Neural Network.

The classifier takes the input and then classifies them as positive and negative emotions. These data are then passed onto another classifier for further classification of positive and negative emotions. The positive features are then classified as enthusiasm, fun, love, happiness, neutral, relief, and surprise. The negative features are classified into anger, boredom, hate, emptiness, sadness, and worry. The classifier then predicts the output of the test input, which provides the results of the model. The text comments from the tweets undergo pre-processing since it contains URL id. Since we tend not to think about any address, we eliminate all the URLs and avoid all unwanted areas. These processes are done in pre–processing stage.

## 4. METHODOLOGY

### 4.1 Dataset

In social networking service, Twitter is a real time messages that lets its users to post called tweets. Tweets have many unique characteristics. Twitter, with nearly 600 million users and over 250 million messages per day, has rapidly turned into a gold mine for organizations to monitor their reputation and brands by extracting and analyzing the sentiment of the Tweets posted by the public about their remarks, markets, and other contenders. Performing Sentiment Analysis on Twitter is complicated than doing it for large reviews. This is because the tweets are very short and mostly contain slangs, emoticons, hash tags and other twitter language [16].The data consists of a large number of tweets collected from the Kaggle repository and Twitter. The Twitter API is used to create a Twitter application and get authorization from tweets. The collected tweet data is in the form of positive as well as negative. Training and Testing dataset consist of both positive and negative tweets.

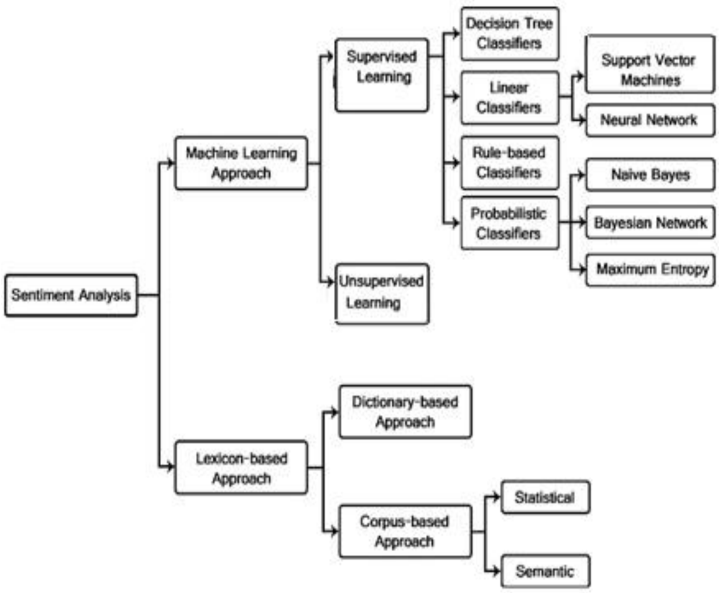


**Fig.1**: Sentiment Classification Technique[1]

### 4.2 Preprocessing

Messages from Twitter are too informal and have different styles of using tweets based on the nationality, origin, age, and gender of the user. Therefore raw Twitter information must be standardized and prepare a formal dataset that can be effectively learned by different classifiers. Most of the preprocessing techniques are generic and can be used in various applications except Sentiment Analysis. Several pre-processing methods such as removal of URL, usernames, hashtags, character normalization, special character removal like punctuations, numbers, special character, lower casing steps have been applied to standardize the dataset and decrease its size.

### 4.3 Feature Extraction

Terms Presence and Frequency, Parts of Speech (POS), Opinion words and phrases, Negations are some the features used for feature extraction process. Individual words like specific unigrams, bigrams and n-grams words with their frequency counts are either uses the term frequency weights or gives binary weighting to the words. The important indicators of opinions of POS are used for finding descriptive words or adjectives from the content. Opinion and phrases words generally used to express opinions like good or bad or hate. Some phrases are express opinions without using opinion words. Generally a document or sentence expresses one opinion orientation like positive or negative about its subject matter. The presence of negative words might change the opinion orientation like not beautiful is equivalent to ugly.

### 4.4 Classification

Sentiment Classification is the binary classification which deals with a small number of classes. Sentiment classification is one of the simple tasks compared to text auto-categorization. While Opinion mining presents various extra tasks other than sentiment polarity detection. The sentiment training set consists of raw tweets labeled positive, negative and neutral. In an effort of making sentiment classifiers, different methods are compared.

#### 4.4.1 Decision Tree

A decision tree is a type of classifier in which each hub or node refers to a test on an attribute of a dataset and its off-spring refer to the outcomes[18]. Decision tree model is applied on test data for node test information. The best test condition or choice has P to be made for each node in the tree. GINI factor is used to select the ideal split. For a given hub or node t,

$$GINI(t)=1-\sum j\ [\ P(j|t)] \qquad (1)$$

where (P(j|t)) is the general recurrence or relative frequency of class j at node t.

#### 4.4.2 Word Embedding

A machine can only understand numbers that convert text to numbers. But word embedding techniques is for converting text into vectors. The word embedding techniques are used to represent words in mathematically. One Hot Encoding, TF-IDF, Word-to-Vec, Fast Text are frequently used Word Embedding methods. One of these techniques in some case is preferred and used according to the status, size and purpose of processing the data.

The word embedding tool uses both Bag-of-Words model and the skip-gram model is used for creating a vector representations of words [19]. Using Neural Networks and Vectorization of text to numbers, word-to-vec is a well-liked approach to Natural Language Processing.

With the help of NLP technology, word optimization, learning, and data correctness can be achieved. The BOW model performs more accurately than the skip-gram model for finding frequent words. In a prior investigation, it was found that word vectors with semantic links enhanced the NLP process of information retrieval and machine translation [20].

#### 4.4.3 Random Forest

The combination of learning algorithm for classification and regression is called Random Forest. Based on the combined decisions of those trees, Random Forest generates a sizable number of decision tree models. For a large number of tweets with the individual assessment marks as sentiment labels such as $x_1, x_2, \dots x_n$, packing $y_1, y_2, \dots y_n$ repeatedly and selects a random sample like $X_b, Y_b$ with substitution. Every arrangement tree $f_b$ is ready to use a different arbitrary example $(X_b, Y_b)$, where b is a number between 1 and B. Finally, a majority vote is cast on these B-trees' forecasts. Using Scikit-Random Forest Classifier Learn's from sklearn. ensemble, a random forest method can be implemented.

#### 4.4.4 Support Vector Machine

Support vector machine is commonly referred to as binary linear classifiers and non-probabilistic. When the feature space is large, this method is recommended for content categorization [21]. The main goal of this classifier is to identify the maximum-margin hyper-plane that separates the points with $y_i = 1$ and $y_i = -1$ for a training set of points $(x_i, y_i)$ where x is the feature vector and y is the class. The equation of the hyper-plane is as follows:

$$W_{x+b=0} \qquad (2)$$

where w is a vector normal to the hyper-plane and b is an offset. Support Vector Machine can be run with both unigram as well as the combination of unigram and bigram.

#### 4.4.5 BERT

BERT is also known as the Transformers' Bidirectional Encoder Representations was published at the end of 2018 [22]. Pre-training language representations have been created using the BERT model, which NLP experts can access and use without charge. In deep learning model, Transformer that is primarily used by BERT to identify the contextual relationships between words or sub-words in the text. The basic design of Transformer's consists of two distinct mechanisms: an encoder that reads text input and a decoder that creates a task prediction. BERT only needs the encoder mechanism because it aims to develop a language model.

Contrary to directional models, which read the input sequentially, the Transformer encoder reads the full input sequence only once (right to left or left to right). Therefore, it is referred to as bidirectional. This feature enables the model to comprehend a word's meaning based on all of its neighbors. In some test cases, Transformers model returns better output than the Google Neural Machine. The encoding portion is made up of a stack of encoders. A stack of identical counted decoders makes up the decoding portion.

The encoders are all comparable in terms of construction. The inputs of the encoder first travel through a self-attention layer, which enables the encoder to view the other terms in the input expression when a given word is encoded. Each of them is divided into two sub-layers. The outputs of the self-attention layer serve as the inputs for the feed-forward network. Since the decoder contains both levels, the attention layer in between them enables it to focus on the pertinent passages in the input paragraph. Self-attention enables to look at specific locations in the input text for indications that may aid to contribute to an improved encoding of the word as the model processes each word. Finding the Value, Key, and Query matrices comes first. The outputs of the self-attention layer are computed as:

$$Z = softmax\left[\frac{QK^T}{\sqrt{d}}\right] \tag{3}$$

Where K is the key matrix, Q is the query matrix and d: is the dimension of the key vector. Pre-trained BERT model is used from the hugging face transformer library for PyTorch.

#### 4.4.6 Bidirectional Network

Hochreiter and Schmidhuber first developed LSTM units in 1997 [23] to address the vanishing gradient issue. The fundamental idea is to put in place a reliable gating mechanism that will control how much the LSTM units retain the derived features of the fresh data input and maintain the previous state. The input gate and its subsequent weight matrices $W_{ui}$, $W_{vi}$, $W_{ci}$, and $b_i$; the forget gate and its subsequent weight matrices $W_{uf}$, $W_v$, $W_c$, and $b_f$; and the output gate and its subsequent weight matrices $W_{uo}$, $W_{vo}$, $W_{co}$, and $b_o$, where $p_{i1}$ denotes the current state of the cell and $v_{i1}$ state produced by the preceding stage. The following equations represent how to choose whether the output state created immediately or later, take inputs, and forget the stored memory.

#### 4.4.7. Convolutional Neural Network

The process of convolution neural network involves acquiring input data and selecting a set of characteristics from it. Data must be first preprocessed before being converted to a vector format and added to a convolution layer. Convolution layer output is pooled using max pooling is the most popular technique. Dropout is applied after pooling to improve accuracy [24]. Create a new feature; a convolution algorithm is applied to filter the input while keeping a window of m words. For produce feature maps, filter is expanded to any window of words

The generated feature maps are then put through a max pooling process where the maximum value of f = max { f } is used. For each feature map, the most pertinent feature with the highest importance needs to be retrieved. Pooling method is naturally addresses sentences with different lengths. These features are subsequently passed to a fully connected softmax layer, the penultimate layer, whose output is non-normalized and corresponds to a probability distribution across predicted labels. The pre-trained word vectors can either be left static while the model's other parameters pick up new information, or can fine-tune them after a few training epochs once the rest of the model has picked up some valuable information. With a restriction on the l2-norms of the weight vector, dropout is utilized for regularization on the penultimate layer.

## 5. RESULTS AND EVALUATION

The performance of sentiment classification can be evaluated by using four indexes calculated as the following equations:

$$Accuracy \; \frac{(TP)+TN)}{(TP+TN+FP+FN)} \quad (4)$$

$$Precision = \frac{(TP)}{(TP+FP)} \quad (5)$$

$$Recall = \frac{(TP)}{(TP+FN)} \quad (6)$$

$$F1 = \frac{(2 \, x \, Precision \, x \, Recall)}{(Precision+Recall)} \quad (7)$$

Comparing the three trained classifiers, the performance of the classifiers using 1,600,000 tested tweets from the sentiment140 dataset.

The Precision, Recall, F1-score, and Accuracy of the BERT, LSTM and CNN model are shown in Table I. In BERT-base-uncased model for training, with bias weight decay set to zero, Layer Norm weights and bias set to 0.001, and all parameters set to 0.001 and fine-tuned for 50 epochs with an 8-batch batch size to operate flawlessly.

**Table – I : Comparison of BERT, LSTM, CNN**

| Model | Precision | Recall | F1 | Accuracy |
|---|---|---|---|---|
| BERT | 0.64 | 0.65 | 0.64 | 64.50 |
| LSTM | 0.60 | 0.62 | 0.61 | 60.05 |
| CNN | 0.59 | 0.61 | 0.60 | 59.20 |

Similarly, In the Bidirectional Attention-based LSTM's performance using 50-dimensional glove embeddings and a 50-dimensional hidden dimension, fine-tune the embeddings after a few epochs when the other layers start picking up important information. In this model trained 50 epochs with a batch size of 16 and used Adam with weight decay and Cross Entropy as the optimizer and loss function to compute gradients and perform back-propagation[16].

The performance of the Convolutional Neural Network-based sentiment classifier with Glove embeddings of dimension 50 with a window size of [3,4,5], and 128 filters in addition to the BERT and Bi-Attentive LSTM classifiers. Adam with weight decay and Cross Entropy, coupled with fine-tuning of the embedding layer, serve as the optimizer and loss function, respectively.

## 6. CONCLUSION

Reduction of data noise and improve accuracy of a model, three well-known deep learning models are studied for sentiment analysis of tweets. Pre-trained Wikipedia language model and the book corpus, which provide a clearer knowledge of the English language and BERT perform better than compared to other classifiers. .

## REFERENCES

[1] Amit G. Shirbhate, Sachin N and Deshmukh, " Feature Extraction for Sentiment Classification on Twitter Data " in International Journal of Science and Research Vol. 5 , 2016

[2] A. Agarwal, F. Biadsy, and K. Mckeown, "Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams," in Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), pp. 24–32, 2009.

[3] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," arXiv preprint cs/0409058, 2004.

[4] S. A. El Rahman, F. A. AlOtaibi, and W. A. AlShehri, "Sentiment analysis of Twitter data," in 2019 International Conference on Computer and Information Sciences (ICCIS), pp. 1–4, IEEE, 2019.

[5] S.-M. Kim and E. Hovy, "Determining the sentiment of opinions," in COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics, pp. 1367–1373, 2004.

[6] Hutto C.J., Gilbert E., VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text, AAAI, 2014.

[7] Qamar, A., Alsuhibany, S. and Ahmed, S. (2017). Sentiment Classification of Twitter Data Belonging to Saudi Arabian Telecommunication Companies. (IJACSA) International Journal of Advanced Computer Science and Applications, [online]

[8] R. M. Duwairi and I.Qarqaz, "A framework for Arabic sentiment analysis using supervised classification", Int. J. Data Mining, Modelling and Management, Vol. 8, No. 4, pp.369-381, 2016.

[9] Hossam S. Ibrahim, Sherif M. Abdou, Mervat Gheith, "Sentiment Analysis For Modern Standard Arabic And Colloquial", International Journal on Natural Language Computing (IJNLC), Vol. 4, No.2, pp.95-109, April 2015.

[10] Assiri, A., Emam, A. and Al-Dossari, H. (2016). Saudi TwitterCorpus for Sentiment Analysis. International Journal of Computer and Information Engineering, [online] Available

at:http://waset.org/publications/10003483/saudi-twitter-corpus-for- sentiment-analysis [Accessed 1 Mar. 2018].

[11] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "Nrc-Canada: Building the state-of-the-art in sentiment analysis of tweets," arXiv preprint arXiv:1308.6242, 2013.

[12] J. Barnes, R. Klinger, and S. S. i. Walde, "Assessing state-of-the-art sentiment models on state-of-the-art sentiment datasets," arXiv preprint arXiv:1709.04219, 2017.

[13] S. Niklander and G. Niklander, "Combining sentimental and content analysis for recognizing and interpreting human affects," in International Conference on Human-Computer Interaction, pp. 465–468, Springer, 2017.

[14] L. M. Rojas-Barahona, "Deep learning for sentiment analysis," Language and Linguistics Compass, vol. 10, no. 12, pp. 701–719,2016.

[15] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS '13), pp. 3111–3119, December 2013.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[17] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in Advances in neural information processing systems, pp. 649–657, 2015.

[18] H. Kim and Y.-S. Jeong, "Sentiment classification using convolutional neural networks," Applied Sciences, vol. 9, no. 11, p. 2347, 2019.