JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Sentiment analysis of short text using KERAS

Mr. Mithun M. Patil, Student, Department of Computer Engineering, TEC, Nerul
Mr. Vishwajit B. Gaikwad Professor, Department of Computer Engineering, TEC, Nerul

***Abstract*- Data available online in the form of text coming from various sources such as social media comments, product reviews, customer queries, search engine queries etc. is huge source of information, and extracting required knowledge out of it is valuable to the business. However processing and inferring meaningful information from this unstructured data is highly challenging task. First and foremost reason is data obtained from social networking comments, customer reviews are usually grammatically incorrect. Further it lacks sufficient statistical information to support many state-of-the-art approaches. Finally they are complex, ambiguous with misspelled words and are generated in an enormous volume, which further increases the difficulty to handle them. After studying multiple methods proposed recently in the field of text analysis it is observed that in order to infer the actual meaning of short text it is essential to have semantic knowledge. This work has proposed a prototype system that uses deep neural network for text processing. The proposed system has two phases namely Model building and Live testing. In first phase, KERAS sequential model is built and trained on YELP dataset of business reviews. In Live testing the user input query is processed in real time to derive its semantics and sentiment which can be either positive, negative or neutral. For getting semantic information the proposed method uses Simple Lesk algorithm and the sentiment is derive using KERAS model built in first phase. Proposed method is tested on Ebay customer review data and results are compared with some of the state-of-art methods namely TextBlob, VADAR analysis and SWN analysis. The results show our method is more effective than Sentiwordnet analysis, almost similar as VADAR analysis but lesser effective as compared to Textblob.**

***Keywords*- Text processing, Simple Lesk, KERAS Sequential model.**

I. INTRODUCTION

In today's world everyone is connected through internet as the smart phones and internet have become so chip that everyone could afford it. With the increasing use of internet, the amount of data generated in the form of short text is enormous. The sources are chatting applications (Whatsapp, Telegram etc), social networking sites (Face book, Twitter, Instagram), online customer reviews, internet blogs, search engine queries, news titles etc. Analyzing and accurately deriving the useful information is very crucial to many business applications such as e-commerce, spam filtering, web search engines, chatbots etc. However analyzing short texts is very difficult and challenging task. Primary reason is short texts usually do not follow language grammar. It may be ambiguous, noisy with random word placement. Further in today's fast-changing world it is generated in huge volume which makes it complex and time consuming to perform tasks such as information extraction, clustering and classification. These challenges give rise to significant amount of ambiguity and make it extremely difficult to handle and process available data. Many text analytics approaches are proposed recently but most of them face challenges mainly due to lack of sufficient statistical information. Consider polysemy of word "apple". It has different meanings such as a fruit, a tree, a company or a brand. Due to the lack of contextual information, these ambiguous words make it extremely hard for computers to understand short texts.

Typically, there are three phases in understanding short text: segmentation, type detection and sentiment analysis. Text segmentation is to break input text into smaller terms which can a word or a phrase. Type detection is to attach meaningful type to each term within input text. Generally POS taggers determine lexical types based on grammatical rules. These approaches are inapplicable as the short texts usually don't follow grammar and lacks sufficient contextual information. Further traditional POS tagging methods cannot distinguish semantic types which, however, are very important for sentiment analysis. In instance disambiguation meaningful labels or types are assigned to each term. These concepts are derived from domain ontology. Sentiment analysis, also referred to as opinion mining, is an approach to NLP that identifies the emotional tone behind a body of text. It helps organizations to determine and categorize opinions about their products, services, and ideas. Organization can use Sentiment analysis to gather insights from complex and unstructured data that comes from online sources such as customer reviews, emails, blog posts, support tickets, web chats, social media channels, forums and comments. In addition to identifying sentiment, opinion mining can extract the polarity or the amount of positivity and negativity within the text. Furthermore it can be applied to varying scopes such as document, paragraph, sentence and sub-sentence levels.

Although the three steps for short text understanding looks straight forward there are many challenges and new approaches must be introduced to tackle these challenges. Short texts are usually noisy, informal and error-prone. It contains abbreviations, nicknames, misspellings etc. For example, "New York city" is sometimes referred as "nyc". This calls for the vocabulary to incorporate as much information about abbreviations and nicknames as possible. Meanwhile, extracting approximate terms is also required to handle spelling mistakes in short texts. Next challenge is ambiguous type where a term can belong to several types, and its best type in a short text depends on context semantics. For example, "watch" in "watch price" refers to wrist watch and should be labeled as instance, whereas in text "watch movie", it is a verb. Short texts are generated in a large volume as compared to whole documents. For example, latest statistics indicate Google now processes over 40,000 search queries every second on average, which translates to over 8.5 billion searches per day and around 3 trillion searches per year worldwide. Twitter is generating around 6000 tweets every second which corresponds to 500 million tweets per day. Therefore, a feasible method for short text processing should be able to handle short texts more effectively and efficiently. However, a short text can have multiple possible segmentations, a term can be labeled with multiple types, and an instance can refer to hundreds of concepts. Hence, it is extremely difficult and time consuming to eliminate these complexities and achieve the best semantic interpretation for a short text.

II. LITERATURE SURVEY

The present work in text analysis is mainly focused on tokenization which spits input text into set of terms or tokens and assigns part-of-speech tags [1][2][3][4][5][6]. For text segmentation various vocabulary based approaches [2][3][4] are proposed which are using online knowledge bases and dictionaries to extract terms. Longest cover method is one of the vocabulary based method which searches for longest matching term in dictionary to segment input text. Chen et al [2] proposed sentiment analysis of twitter data using an unsupervised method of named entity recognition (NER) which utilizes Wikipedia and web corpus for segmentation. Hua et al. proposed trie-based framework [1] which uses graph to represent terms which are candidate

for segmentation and their relationship. One of the common drawbacks of existing methods for text segmentation is they only consider lexical features and ignore the semantics within the segmentation. Statistical methods for segmentation calculate occurrences of two terms together in corpus. N-gram [2][3] is one of the statistical model which calculates frequencies of two or more words occurring together in corpus to decide whether those words can be treated as a term. Semantic hashing [4] is another approach which represents text into binary code which is then used for clustering. However for short texts such approach can yield incorrect information sometimes because of noisy nature.

In Part-of-speech tagging appropriate lexical types are assigned to individual terms based on their meaning and context. It can be done using grammatical methods which uses predefined rules or statistical approaches [1] which uses models trained on large corpus. Rule based approach incurs high cost of constructing production rules however gives stable results. Whereas statistical models use learned statistics instead of tagging rules to assign tags, here the results are unstable. Both the approaches assume that terms are correctly arranged in given input which may not be the case in short text. Song et al. [5] proposed a method of short text understanding by using a popular knowledge base Probase [5] for getting real world concepts and uses Bayesian inference for building words concept vector. However most of the existing knowledge bases are limited in scale and scope. Further most of them do not consider content semantics.

Ming et al [7] proposed a long short term (LSTM) based recurrent neural network model which recognize text emotion by deriving two word vectors semantic and emotional. It can detect seven distinct emotions categories as anger, anxiety, boredom, happiness, sadness, disgust and surprise. LSTM models overcome the drawback of traditional recurrent neural networks that is can't learn long distance dependent information. Jin et al [8] proposed bag of words model to process short texts for duplicate detection. It has used Word2vec to derive word vectors and Simhash algorithm is used to compare sequences using hamming distance.

Table 1: Comparison table of various methods proposed for short text understanding

SR No.	Author	Method	Outcome	Limitations	Year
1	Y. Song, H. Wang, Z. Wang, H. Li, W. Chen [5]	Short text conceptualization using Probase	It finds named entities in input text and assigns it meaningful labels.	Applicability is limited. Does not focus on word semantics.	July 2011
2	C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee, [2]	Named entity recognition in targeted Twitter stream	It uses Wikipedia and web corpus to segment tweets using n-gram method and does NER using unsupervised model	Applicability is restricted to tweets only.	August 2012
3	Z. Yu, H. Wang, X. Lin and M. Wang [4]	Short text clustering using Deep Neural Network	It uses DNN to convert input text into binary code and then two texts are compared using their binary code to cluster similar texts.	Heavy computation is required. Model output totally depends upon quality of testing set.	Feb 2016
4	W. Hua, Z. Wang, H. Wang, K. Zheng and X. Zhou [1]	Semantic analysis of short text using online knowledgebase and web corpus	Segmentation is done by applying Monte Carlo algorithm on term graph, followed by POS tagging using Standford tagger	Totally dependent upon online knowledge base. Computation of co-occurrence network is very complex and need much time and space.	March 2017
5	M. Su, C. Wu, K. Huang, Q. Hong [7]	Text emotion recognition based on word vector	It first extract semantic word vector and emotional word vector using word2vec model and auto encoder respectively. The concatenated vector is analyzed using LSTM to recognize text emotion.	Results are derived from limited amount data. Need to improve accuracy.	May 2018
6	J.Yang, G. Huang, B. Cai [10]	Short text clustering using TRTD	Topic representative terms are discovered by individual occurrence frequency and co-occurrence frequency	Simple yet effective method of text clustering. Discover topic terms based on high frequency count which may not be the case always.	July 2019
7	R.Man, K.Lin [11]	Sentiment Analysis Algorithm Based on BERT and Convolutional Neural Network	Article feature extraction using BERL and Convolutional neural network	Heavy computation is required. Model output totally depends upon quality of testing set.	April 2021

III. PROBLEM STATEMENT

Analyzing textual data available online in multiple forms is crucial to many e-commerce and other business processes. It is important for organizations to accurately analyze their customer reviews, social networking posts, news and chatbots queries which are in the form of short texts, in order to better understand customer needs and gain competitive advantage. However processing this unstructured, complex and huge data is highly challenging task

as it lacks contextual information. Further short texts are noisy, may contain abbreviations and ambiguous words which makes it extremely difficult to infer semantic meaning out of it. As a result, traditional natural language processing tools, such as part-of-speech tagging, dependency parsing cannot be applied efficiently to short texts.

Given a short text s written in a natural language, we generate a semantic interpretation of s represented as a sequence of typed-terms namely

$\overline{s} = \{\overline{t}_i | i = 1, \dots, n\}$

And from semantic knowledge we determine the sentiment of input text which can be either positive, negative or neutral.

E.g.

Input sentence: "went bank to deposit money"

Output: Went [verb], bank [noun –financial institution], deposit [verb], money [noun]

Sentiment: Neutral

IV. PROPOSED SYSTEM

Many approaches have been proposed recently to enable short text understanding. These methods, however, have their own limitations due to limited context available. Without knowing the word semantics and distribution of the senses, it is difficult to build a model to choose appropriate semantic tag for a word in a context. The work most related to ours are conducted by Wen et al. [1] , Katekar et al. [12], Apoorva et al[13] and Rashid et al[14], which derives the semantic of text using statistical models. The system developed by Katekar [4] uses HMM for text segmentation and type detection and K-means clustering for grouping the similar texts. Others used SentiWordNet [13], TextBlob [14] methods for sentiment analysis.

In this work, it is observe that other terms, such as verbs, adjectives, and attributes can also help with instance disambiguation. Short texts do not always observe the syntax of a written language. As a result, traditional natural language processing tools, such as part-of-speech tagging to dependency parsing fail to process short texts. Short texts usually do not contain sufficient statistical signals to support many state-of-the-art approaches for text mining such as topic modeling [13][14]. Further they are more ambiguous and noisy, and are generated in an enormous volume, which further increases the difficulty to handle them.

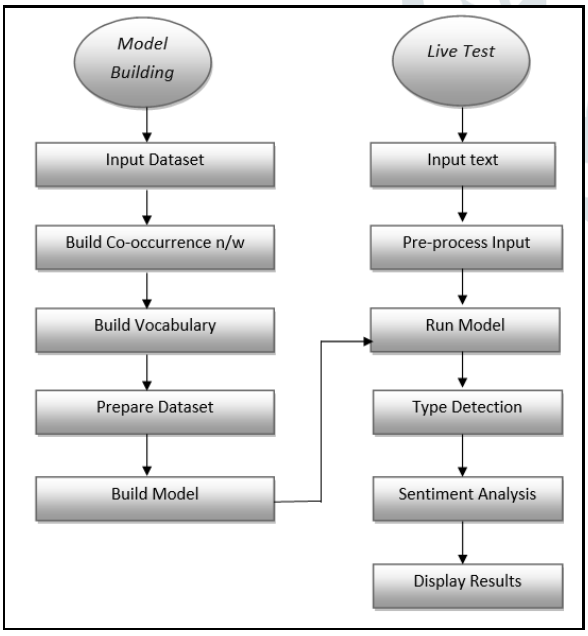

Fig 1: Flow diagram of short text analyzing method

Fig. 1 illustrates the proposed system framework for short text understanding. The proposed has two phases namely Model Building and Live Testing. In Model building, Yelp dataset with 44025 records depicting the customer reviews about different business processes is used. Initially the dataset is pre-processed by removing English stop words and applying Snowball stemmer. Bigrams are derived from the cleansed data and most frequently co-occurring words are grouped together forming co-occurrence network. Then a vocabulary index is built using W2V model [7] based on word occurrence frequency. Yelp dataset is broken into training set containing 35220 records and testing set containing 8805 records. The vocabulary index is used for converting reviews in textual format in training and testing set into word embeddings as the machine learning model understands only numbers. Label Binarizer is used to encode labels in training and testing set into sequence vectors. Finally the input reviews in the form of embedding vector and output labels in the form of sequence vectors are fed to KERAS sequential model. The model is trained and fitted with 20 EPOCHS and batch size of 512 records yielding training accuracy of 78.41%.

In Live testing part, the user input text is processed in real time to derive its semantic knowledge and sentiment. Initially the input text is pre-processed to correct the misspelled words. Then the text is transformed into word embedding using tokenizer and fed to model to predict its sentiment. Words in input text are disambiguated using Simple Lesk algorithm [14] to derive its part-of-speech tag and contextual meaning.

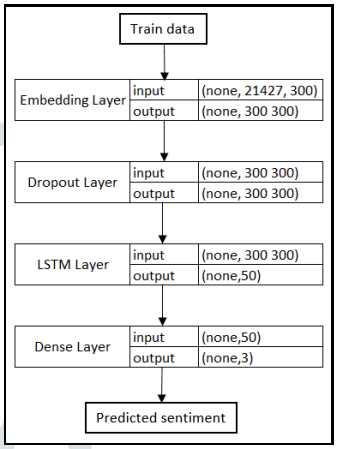

Fig.2: KERAS sequential model

Fig. 2 depicts the structure of KERAS sequential model. It is a type of RNN with four layers namely Embedding layer, Dropout layer, LSTM layer and Dense layer. Embedding layer is first hidden layer having three arguments, input dimension or vocabulary size (21427), output vector dimension (300) and input sequence length (300). Embedding matrix is prepared using vocabulary index and it is used to initialize the weights of Embedding layer. Second layer is dropout layer which randomly sets input units to 0 with the rate of 0.2. It means in every forward pass, around 20% of input neurons will be skipped while deriving output of next layer. Dropout is a regularization technique which is used to prevent the problem of model over fitting. Third layer is LSTM which is mainly used to learn the long-term dependencies in sequence data. It has 50 hidden units used to store information from all previous time steps. Final layer is dense layer which is fully connected layer with 3 neuron. It performs matrix-vector multiplication to get 3-dimensional prediction vector as we have 3 labels in classification namely positive, negative and neutral.

V. Result Analysis

The performance of proposed method is evaluated via Accuracy, Precision, Recall & F1 Score metrics. Below results analysis shows the effectiveness of sentiment analysis by multiple methods namely Textblob Analysis, VADAR analysis, SentiwordNet Analysis and proposed method based on KERAS sequential model using LSTM.

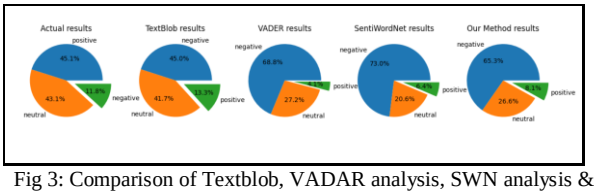

Fig 3: Comparison of Textblob, VADAR analysis, SWN analysis & proposed method based on KERAS model

Fig. 3 shows the sentiment count predicted by various methods on Ebay dataset of customer reviews. We have compared our method with other three methods namely Textblob, VADAR and SentiwordNet analysis. Three labels

namely positive, negative and neutral are used to denote the sentiment of input text. Efficiency of these methods is compared using precision, recall, F1 Score and accuracy metrics.

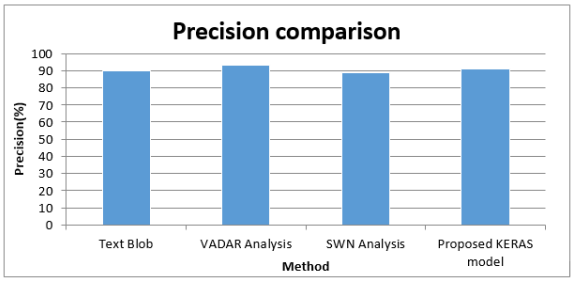

Fig 4: Precision comparison of Textblob, VADAR analysis, SWN analysis & proposed method based on KERAS model

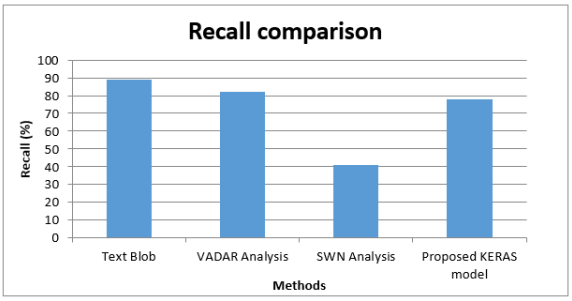

Fig 5: Recall comparison of Textblob, VADAR analysis, SWN analysis & proposed method based on KERAS model

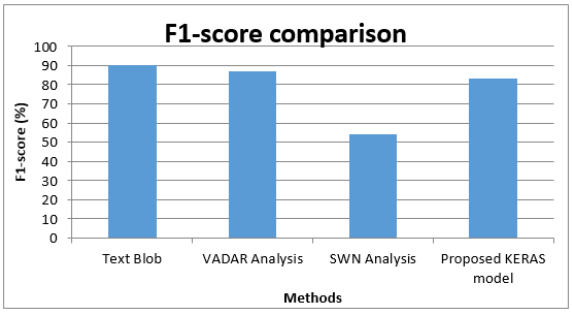

Fig 6: F1-score comparison of Textblob, VADAR analysis, SWN analysis & proposed method based on KERAS model

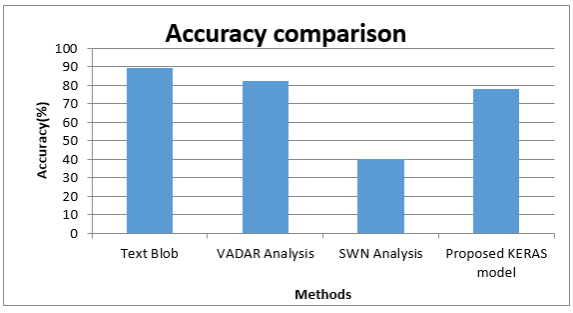

Fig 7: Accuracy comparison of Textblob, VADAR analysis, SWN analysis & proposed method based on KERAS model

Table 2: Results comparison using performance metrics

Method	Precision	Recall	F1-score	Accuracy
Text Blob	90%	89%	90%	89%
VADAR Analysis	93%	82%	87%	82%
SWN Analysis	89%	41%	54%	40%
ProposedKERAS model	91%	78%	83%	78%

Table 2 depicts the precision, recall, F1-score and accuracy score of all the methods in comparison. Results show Textblob analysis is best among all the methods in comparison. It has highest accuracy (89%) and F1-score (90%). VADAR analysis has highest precision (93%). It tells of all the reviews that are labeled as positive, how many are actually positive. Textblob method has highest recall (89%) which tells of all the reviews that are actually positive, how many we labeled positive. Proposed method is more effective than Sentiwordnet Analysis in terms of all the four metrics. Proposed method has Precision almost similar as Textblob & VADAR analysis however the Recall is lower. F1-score is usually more useful than accuracy, especially if you have an uneven class distribution. The results show our method is more effective than Sentiwordnet analysis, almost similar as VADAR analysis but lesser effective as compared to Textblob. Hence it needs more improvements to compete with other state-of-art methods.

VI. CONCLUSION

This paper studies existing work in the field of semantic and sentiment analysis of the short text data available online in the form of tweets, social networking posts, customer reviews, comments, search engine queries etc. Here text under analysis is short with limited number of words. Multiple rule-based as well as statistical methods have been proposed recently in the field of text processing but all have their own limitations due to multiple challenges. Primary challenges in text analysis are lack of contextual information, noisy or misspelled words and enormous volume.

In this work a generalized framework is proposed to analyze semantic and sentimental knowledge of short text data. An effort is made in anticipation of getting an alternative method to understand short texts effectively, which exploits semantic knowledge. The proposed method is divided into two phases Model building and Live testing. In Model Building, initially the input dataset is pre-processed to remove URLs and stop words. The misspelled words are corrected and words are reduced to their stems using Snowball stemmer. Co-occurrence network is then built using more frequently co-occurring terms and a vocabulary index is built using W2V model. The proposed method uses LSTM-based KERAS sequential model, which is a deep learning network, for determining sentiment of short text. The model is trained on Yelp dataset with 44,025 records of customer reviews about various business processes and sentiments classified as positive, negative and neutral. In live testing, user input text is processed in real time to determine its semantics and sentiment. For semantic type detection proposed method uses Simple Lesk algorithm which assigns the meaningful types and contextual description to the words in input. For determining the sentiment, it uses KERAS sequential model trained on YELP dataset.

For performance analysis, Ebay dataset of customer reviews is taken for validation testing. The results of proposed knowledge-intensive approach are compared with existing state-of-art methods namely: Text Blob analysis, VADAR analysis and Sentiwordnet analysis. The results show our method is more effective than Sentiwordnet analysis, almost similar as VADAR analysis but lesser effective as compared to Textblob. Hence it needs more improvements to compete with state-of-art methods. To improve the accuracy of proposed method further, it is advisable to have more concrete dataset. In future, multi-language model needs to be build which will be both effective and efficient in sentiment analysis.

REFERENCES

[1] W. Hua, Z. Wang, H. Wang, K. Zheng and X. Zhou, “Understand Short Texts by Harvesting and Analyzing Semantic Knowledge”, IEEE Transactions on Knowledge and Data Engineering, vol. 29(3), March 2017, pp. 499-512.

[2] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B. Lee, "Twiner: Named entity recognition in targeted twitter stream,” in Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, USA, Aug 2012, pp.721–730.

[3] M. Hagen, M. Potthast, B. Stein, and C. Brautigam, "Query segmentation revisited", in Proceedings of the 20th International Conference on World Wide Web, New York, USA, 2011, pp. 97–106.

[4] Z. Yu, H. Wang, X. Lin and M. Wang, "Understanding Short Texts through Semantic Enrichment and Hashing", IEEE Transactions on Knowledge and Data Engineering, vol. 28(2), Feb 2016, pp.566 - 579

[5] Y. Song, H. Wang, Z. Wang, H. Li, W. Chen, "Short Text Conceptualization using a Probabilistic Knowledgebase", IJCAI Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, vol.(3), July 2011, pp.2330-2336.

[6] Z. Wang, K. Zhao, H. Wang, X. Meng, and J. Wen, "Query Understanding through Knowledge-Based Conceptualization", IJCAI, July 2015.
[7] M. Su, C. Wu, K. Huang, Q. Hong, "LSTM-based Text Emotion Recognition Using Semantic and Emotional Word Vectors", IEEE First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia), May 2018
[8] J. Gao, Y.He, X. Zhang, Y. Xia, "Duplicate Short Text Detection Based on Word2vec", IEEE International Conference on Software Engineering and Service Science, Nov 2017
[9] E. Brill, "A Simple Rule-Based Part of Speech Tagger", Proceedings of the third conference on Applied natural language processing, Pages 152–155, March 1992.
[10] J.Yang, G. Huang, B. Cai, "Discovering Topic Representative Terms for Short Text Clustering", IEEE access, vol .7, July 2019, pp. 92037 – 92047
[11] R.Man, K.Lin, "Sentiment Analysis Algorithm Based on BERT and Convolutional Neural Network", IEEE Conference, April 2021
[12] A. Katekar, "Improving the Effectiveness of Short Text Understanding by Using Web Information Mining", in Proc. IEEE 2017 ICCMC, July 2017.
[13] A. Agarwal, V.Sharma, G.Sikka, R.Dhir, "Opinion Mining of News Headlines using SentiWordNet", IEEE conference, Sep-2016.
[14] R.Khan, F.Rustom, K.Kanwal, A.Mehmood, G.Choi, "A. Agarwal, V.Sharma, G.Sikka, R.Dhir, "US Based COVID-19 Tweets Sentiment Analysis Using TextBlob and Supervised Machine Learning Algorithms", IEEE Conference, June-2021.