IJCRT.ORG ISSN : 2320-2882
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

SENTIMENT ANALYSIS USING MACHINE LEARNING AND DEEP LEARNING APPROACHES IN WORD EMBEDDING TECHNIQUES

[1]Arul Jothi S, [2]N. Pandeeswari, [3]Deepika A
[1]Assistant Professor, [2]Associate Professor, [3]Assistant Professor
[1,3] Department of Computer Science and Engineering
[2]Department of Information Technology
[1,3]Nadar Saraswathi College of Engineering and Technology, Theni, Tamilnadu.
[2]PSNA College of Engineering and Technology, Dindigul, Tamilnadu.
[1]aruljothi@nscet.org,[2]pandeeswari@psnacet.edu.in,[3]deepika@nscet.org

Abstract: Sentiment analysis is the task of extracting contextual information from the text. The source of textual data can be from social media posts, comments, reviews, blogs, tweets, news articles etc…These text involves in stop word removal, lemmatization and stemming. After text preprocessing steps, the text has to processed using word embedding techniques. The word has to processed into vectors that can involve into machine learning and neural network approaches. Word2vec and skip gram are the two successful approaches in supervised machine learning and GLoVe in unsupervised machine learning. Recurrent Neural Network(RNN) and Convolution Neural Network (CNN) are the two approaches in deep learning algorithm. Thus the paper discusses the machine learning and deep learning algorithm in word embedding techniques can be used for feature extraction. Then it can be fed into various classifiers to fall under the category of positive, negative or neutral which helps to understand the moral and ethical values of people in various applications such as business, politics, ecommerce and health care industry and in movie reviews. The experimental results show that deep learning approaches produce better results in sentiment prediction when compared with Machine learning.

***Index Terms* - Semantic analysis, word2vec, word Embedding, Machine Learning, Deep Learning, RNN, CNN.**

I. INTRODUCTION

Today, People from all around the world are connected with wide range of social networks. Social networks like Face book, Twitter, Instagram are intensively increased day by day. The textual data is a primary entity which is shared among them. This textual data need to be analyzed and classified called text mining. Text mining brings tremendous changes in various fields, in technology area that has been incorporated in several research fields such as computational linguistics understanding human language (Natural language Processing), Information Retrieval (IR). The huge studies have been made by various researchers, and built various machine learning systems and deep learning system for text analytics.

II. RELATED WORKS

Text analysis in social media towards politics, ecommerce is used to investigate people's sentiments, attitudes, opinions emotions, etc. The main task of text analysis [1] in human languages falls in the field of Natural Language Processing are summarization of text, sentiment analysis, chat bots and language translation. Extracting useful information from the text plays a vital role in text classification and also in Sentiment analysis or opinion mining by analyzing attitudes, feelings, their opinion from the text. Bag of words [6] model also used in sentiment analysis.[2] enhances the security of vectorization using word2vec and cryptDB [4],[5] Word embedding Techniques used in sentiment analysis and also used for software engineering texts.[3][7] Similarity terms expressed in word embedding techniques.[8]Sentiment Analysis using Deep learning approaches are trained with different datasets.Term Frequency-inverse document frequency and word embedding also compared with datasets
Section III describes about the various phases of analyzing social media text.
Section IV discussed about Machine Learning Approaches in Word Embedding Techniques
Section V reveals about the word embedding approaches in Deep Learning Algorithm
Section V Classifiers Performance Metrics such as precision, recall, accuracy measurement has been formulated

2.1. VARIOUS PHASES OF ANALYZING SOCIAL MEDIA TEXT IN ML.

The Fig.1. depicts the various phases of text processing which includes text collection, text preprocessing, text representation, sentiment classifiers, and sentiment prediction.

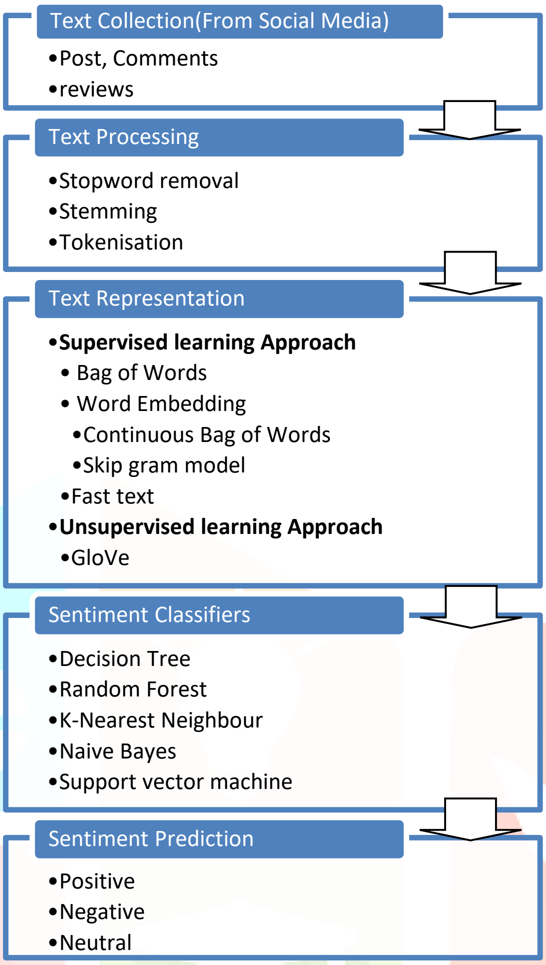


Fig.1.Various phases of text processing

2.1(a)Text Collection

Text is collected from Social Media post, comments and reviews exhibiting different people moral and social values. The websites provide a very powerful medium for communication among individuals that leads to mutual learning and sharing of valuable knowledge. On social media it becomes a common way to not write a sentence with correct grammar and spelling. This practice may lead to different kinds of ambiguities like lexical, syntactic and semantic and due to this type of unclear data, it is hard to find out the actual data order. Extracting features from the unstructured data is a critical task in text analysis. This raw data can be cleaned by preprocessing steps

2.1. (b)Text Processing

Text processing involves steps like stop word removal (removing full stop, comma, punctuation, semi-colon or any symbols), stemming (find out the root word for all similar word), convert the words to lower case, tokenize the sentences into words.

2.1(c) Text Representation

Machine learning and deep learning method involves in text representation. The main role is to convert the tokens into vector (numeric value) representation. The feature extraction from text can be done by this learning approach. Word Embedding is the challenging task in this phases.

2.1. (d)Sentiment Classifiers

The features are extracted by learning approaches and it is given as input to the various classifiers such as Decision Tree classifier, K-Nearest Neighbor, Naïve Bayes, Support vector Machine which involves text classification for better accuracy. The comparisons are made under different classifiers for better results. Nowadays advanced classifier methods such as ensemble classifiers which combines the advantage of different classifiers opts better prediction.

2.1. (e)Sentiment Prediction

Sentiment are categorized with tags such as positive, negative and neutral. The prediction parameters are precision and recall by the better score of false negative and true positive.

III. MACHINE LEARNING ALGORITHM

The tokenization of words can be modeled into matrix representation by the word embedding learning methods. It is generally classified into Supervised learning method and unsupervised learning method.

3.1. Supervised learning:
//input: labeled data and input data
//Process: uses an algorithm
//output: predict the test-input data with trained set of labeled data.

It helps to predict the new data based on the trained set of data using an algorithm below.
Consider the input variable as X and the output variable as Y respectively. The algorithm deals with mapping the function from input to output is of the form Y=f(X). Here the output variable Y can be predicted based on the labeled trained set of X input data.
The learning approaches based on supervised methods falls into three categories:
(i)Count Based-Bag of words.
(ii)Window Based-Word Embedding (CBOW, Skip gram)

3.1(a). Bag of words.

In supervised learning method, the Bag of words (BOW) is the simplest method for generating matrix with individual word and its frequency count. Bag of words also known as vectorization[2]. It is a count based model. Semantic analysis of text in this method is difficult.

```
(a)Algorithm:  Bag of Words /Vectorisation
//input: Tokens (w1,w2,…wn),n is number of words in document
//output: word-count matrix(wij), wij  is Sparse matrix

Read the tokens(w1,w2,…wn) from d (social media text)
For each word wi in d
            Built  the word-count matrix,wij
Word_vocabulary<-wij
```

It generates sparse matrix with meaningless context and it does not preserve the semantic relation. High computation is needed for large collection of text. The order of words is not considered.

3.1(b). Word Embedding

Word Embedding is the effective method for converting high dimension data into lower dimension data preserving semantic relationship. Similar semantic phrases have similar distances in the vector space. Collaborative filtering is the technique to find the similar words with closest neighbor. PCA (Principal Component Analysis) is a method of word embedding which convert higher dimension to single dimension space. It is a window based model.

(i)Word2vec[4] is a neural network approaching method operates under Distributional Hypothesis, map the similar words to numerically embedding vector. It generates dense vector representation for individual word.

```
(b)Algortithm: Word Embedding
Input: Sparse matrix with higher dimension, wij [1xN] and Embedding vector[NxM]
Output: Lower dimension, wij[1xM]

int wordembed (sparse matrix, Embedding matrix)
    For i in range(1,N)
      For j in range (N,M)
         Mij=0
           For k in range(1,M)
             Wij=wij+wik* wkk* wkj
   Return(dense matrix ,wij)
```

(ii)Continuous Bag of words (CBOW) and
(iii)Skip-gram are successive method for vector representation under Distributional Hypothesis,
CBOW[6] generate continuous distributed representation of the context word vectors in the corpus to be located close to one another in the vector space. It can predict the future based on the context. It is faster to train compared to skip gram for frequent words

```
//CBOW Model Approach
(Neural Network Approach)
//Input layer Dimension: one or more context word vector (1xV), V number of words in the vocabulary

//Hidden layer Dimension:
V x E (E denotes size of word embedding and its hyper parameter)

//output layer Dimension:
Predict the current word (1 x E)
```

Skip gram Model also same as continuous bag of words model. It works well for smaller dimension of data and well suited for rare phrases or words

```
//Skip-gram Model Approach
//Input layer Dimension: current word vector
(1xV), V number of words in the vocabulary

//Hidden layer Dimension: V x E (E denotes size of word embedding and its hyper parameter)

//output layer Dimension: Predict the context before and after the current word (1 x E)
```

The models are frequently trained using either a hierarchical softmax function (HS) or negative sampling (NS) for efficiency.Methods like skip-gram perform better on the analogy task, but poorly utilize the statistics of the corpus as they are not trained on global co-occurrence counts.

3.2. Unsupervised Learning:

It helps to learn new data without having labeled data set and appropriate algorithm. It can be used to learn mapping function from input to expected output.

```
// input:  unlabeled data
//Process: uses an algorithm
//output: predict and classify the output into categorization of clustering
```

An unsupervised learning algorithm of GloVe is a salient features of aggregated global word-word co-occurrence vector representation of word. The global matrix factorization method integrated with that of context window based method makes a model of global log bilinear regression model. Dense vector of each vector in Glove is to combine all the words authored by a user, and can also use a vector aggregation function such as average to combine the vectors of all the words in a user's posts.

3.3.FastText

FastText is the character level embedding techniques which is the extension of word2vec. It refined embedding by summing character n-gram vectors with vectors of surrounding words.

IV.DEEP LEARNING TECHNIQUES

Deep learning is the subfield of machine learning which provides the model for complex relation of data. It includes Recurrent Neural networks (RNN) and Convolutional Neural Networks(CNN).

4.1. Recurrent Neural Networks(RNN)

Recurrent Neural Networks(RNN) are mainly used in sentiment analysis for text classification. The sequential data has to be captured with its internal memory. It has the capability to remember the past sequences along with current input to capture the current context rather than individual words. Simple RNN causes the problem of vanishing gradients. It can be overcome by LSTM (Long Short Term Memory unit) with input(4.1),(4.2),(4.3),and output (4.4),(4.5)which has been used in Natural Language Processing. It consists of input gate, an output gate, a forget gate and a cell with given time t. Using Back propagation, the weights for these gates and cell get trained It effectively deals with long dependencies

//GloVe Algorithm
//input:
X=matrix of word-word co-occurrence counts.
X_{ij}=number of times word j present in the context of the word i
$X_i=\sum_k X_{ik}$ number of times any word in the context of the word i
//output: $P_{ij}=P(j/i)= X_{ij} / X_i$ Dense vector for each word
// Process:a vector aggregation function such as average to combine the vectors of all the words in a user's posts.

//LSTM
//Input: Given time t,
$i_t = \sigma(w_{xi}x_t + w_{h_i}h_{t-1} + w_{ci}c_{t-1} + b_i)$ (4.1)
$f_t = \sigma(w_{xf}x_t + w_{h_f}h_{t-1} + w_{cf}c_{t-1} + b_f)$ (4.2)
$c_t = f_t\, c_{t-1} + i_t \tanh(w_{xc}x_t + w_{h_c}h_{t-1} + b_c)$ (4.3)
//Output:
$o_t = \sigma(w_{xo}x_t + w_{h_o}h_{t-1} + w_{co}\, c_t + b_o)$ (4.4)
$h_t = o_t \tanh(c_t)$ (4.5)

4.2. Convolutional Neural Networks

Convolutional neural networks mainly used in image processing and computer vision application and also used in sentiment classification. It is used with 32 filters (parallel fields for processing words) and a kernel size of 8 with an activation function(4.7),(4.8),(4.11). It is followed by Max pooling layer with reduced size(4.12),(4.13). The output can be flattened into long vector which can extracted the features. This layer can use activation function which the value lies between 0 and 1 to represent positive and negative sentiment.

Stage1: Fully Connected layer

//Input C*w*h for k neurons
$|W|$=cwhk+k=(cwh+1)k (4.6)
W=no of elements in weight matrix
w=No.of elements per kernel
h=No.of output channels
k=No.of Bias connections

The operations perform for per image for fully connected having two cases, with pipelined and non-pipelined processors.
In Pipelined Processors
#multiplications per output node=n
#additions per output node=n
#Operations per output node=n+1
#Total ops in layer=(n+1)k (4.7)
In Non pipelined Processor
#Operations per output node=2n
#Total ops in layer=2nk (4.8)

Stage2: 2D convolution
//input Parameters
Padding on width=p_w
Padding on height=p_h
Stride on width=s_w
Stride on height=s_h
Width of output tensor=$\left(\frac{M-w+2Pw}{sw}+1\right)$ (4.9)
Height of output tensor=$\left(\frac{M-h+2Pw}{sh}+1\right)$ (4.10)
//Output

$$Y_{r,c=} = w_{k,0} + \sum_{i=1}^{w} \sum_{j=1}^{h} w_{k,i,j} Xf(r) - i\,, f(c) - j \quad (4.11)$$

#multiplications per output node=n
#additions per node=n

In Pipelined Processor the total operations in layer=k(cwh+1) $\left(\frac{M-w+2Pw}{sw}+1\right)\left(\frac{M-h+2Pw}{sh}+1\right)$

Stage 3: 2D Max Pooling
$Y_{r,c=} = \max(i,j)\ \mathcal{E}(w,h)\{Xf(r) - i, f(c) - j\}$ (4.12)

Output:
#comparisons per location =wh-1
#total operations in layer
=c(wh-1)$\left(\frac{M-w+2Pw}{sw}+1\right)\left(\frac{N-h+2Ph}{sw}+1\right)$ (4.13)

Not in sense that more number of parameter leads to more number of operations.

V.CLASSIFIERS PERFORMANCE METRICS

Confusion matrix is a tool to analyze the performance of correct or incorrect classification based on the metrics of True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) in the measurement of F1 score. F1 score is based on the calculation of precision and recall.

F= (Precision*recall)/ (Precision + Recall)
F1 Score=2 *F

5.1.Precision:
It is measured by the ratio of True Positive and the sum of True Positive and False Positive.
It determines the number of

Precision=TP/TP +FP

5.2.Recall:
It is measured by the ratio of True Positive and the sum of True Positive and False Negative. It shows the correct identification of relevant data

Recall=TP/TP +FN

5.3. Accuracy
It is the metrics measures deals with the ratio of correct predictions to the total number of predictions.
Higher the accuracy rate gives the better prediction.

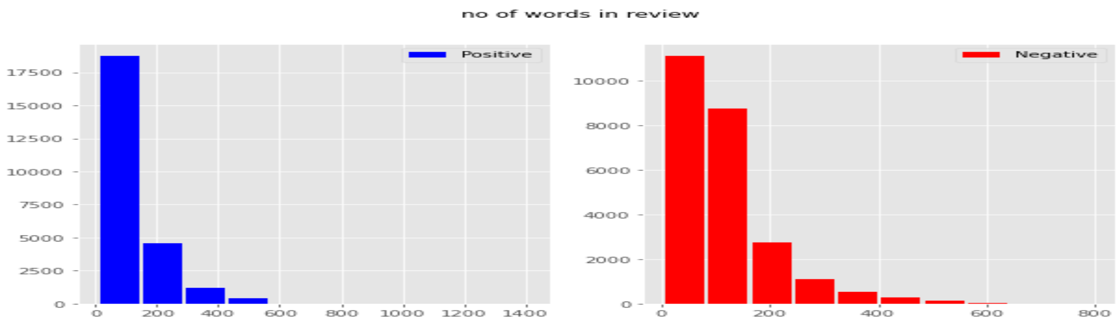
Accuracy=TP+TN/ (TP+TN+FP+FN)

Data sets:
IMDB dataset having 50k movie review for text analytics. This dataset available from kaggle website in the link of https://www.kaggle.com.IMDB dataset is used for binary classification having positive and negative reviews. It is used for machine learning and deep learning algorithms

VI.EXPERIMENTAL RESULTS

Sl.No.	Deep Learning Approaches	Precision	Recall	F1-Score	Accuracy
1	CNN	0.72	0.75	0.76	0.78
2	RNN+LSTM	0.736	0.77	0.79	0.81

Table 1 Testing Results for various ML Approaches



IMDB dataset having positive and negative reviews. In this number of words in a positive and negative reviews has calculated and expressed through chart.

Fig 2. Chart for the number of words in positive and negative reviews

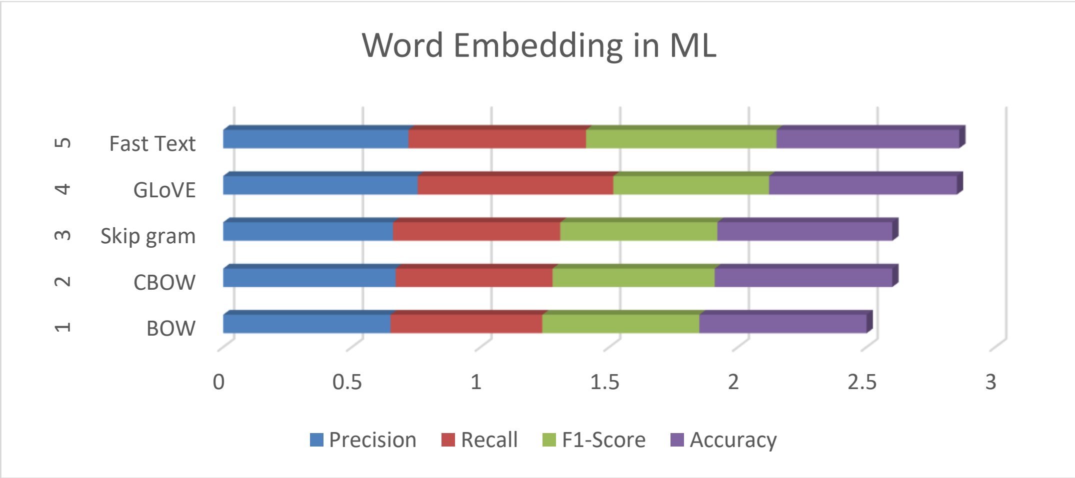
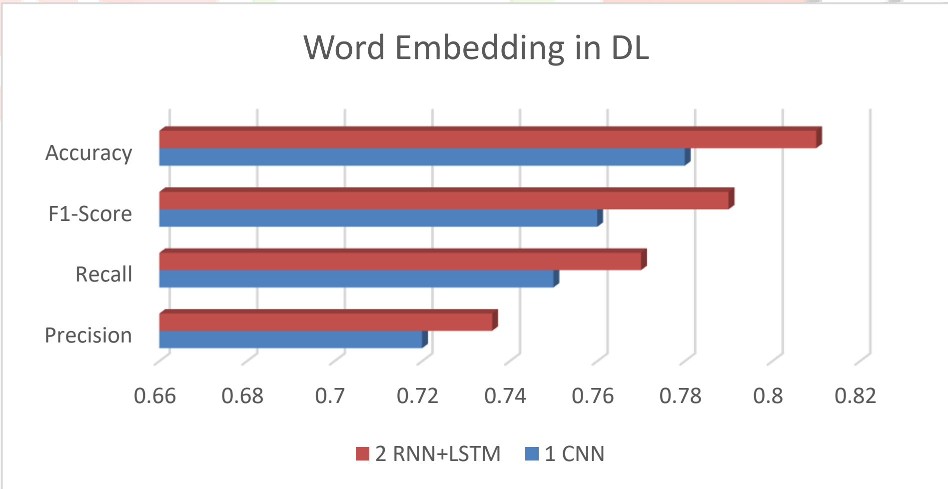


Fig.3 shows Word Embedding in various ML Approaches

Table 2. Testing Results

Sl.No.	Machine Learning Approaches	Precision	Recall	F1-Score	Accuracy
1	BOW	0.65	0.59	0.61	0.65
2	CBOW	0.67	0.61	0.63	0.69
3	Skip gram	0.66	0.65	0.61	0.68
4	GLoVE	0.756	0.76	0.605	0.73
5	Fast Text	0.72	0.69	0.74	0.71

Fig4. Word Embedding in Deep learning Techniques



CONCLUSION AND FUTURE WORK

The results show that Machine learning and Deep learning approaches for the word embedding techniques increases the performance of all the models. RNN with LSTM approach grants a better improvement than GLoVe in this sentiment approach. The future work is to the combined features of GLoVe and RNN, GLoVe and LSTM, CBOW and LSTM will produce better results.

REFERENCES

[1].Nalini Chintalapudi , Gopi Battineni , Marzio Di Canio , Getu Gamo Sagaroa , Francesco Amentaa,"Text mining with sentiment analysis on seafarers", medical documents, International Journal of Information Management Data Insights journal homepage: www.elsevier.com/locate/jjimei,

[2]. Hana Yousuf , Said Salloum ,"Survey Analysis: Enhancing the Security of Vectorization by Using word2vec and CryptDB", Advances in Science, Technology and Engineering Systems Journal Vol. 5, No. 4, 374-380 (2020),

[3]. Navnita Nandakumar Bahar Salehi Timothy Baldwin, "A Comparative Study of Embedding Models in Predicting the Compositionality of Multiword Expressions"

[4].Oscar B. Deho, William A. Agangiba et.al.,"Sentiment Analysis with Word Embedding", Research gate, August 2018 , https://www.researchgate.net/publication/328523241,DOI: 10.1109/ICASTECH.2018.8506717

[5]. Eeshita Biswas, K. Vijay-Shanker, Lori Pollock,"Exploring Word Embedding Techniques to Improve Sentiment Analysis of Software Engineering Texts", May 2019 , https://www.researchgate.net/publication/333389939

[6]. Doaa Mohey El-Din,"Enhancement Bag-of-Words Model for Solving the Challenges of Sentiment Analysis ",) International Journal of Advanced Computer Science and Applications, Vol 7, No. 1, 2016

[7]. Yating Zhang ,Adam Jatowt, Katsumi Tanaka, " Towards Understanding Word Embeddings: Automatically Explaining Similarity of Terms", 2016 IEEE International Conference on Big Data (Big Data)

[8]."Sentiment Analysis Based on Deep learning :A Comparative study", Research gate, March 2020. DOI:10.3390/electronics9030483