Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

An Efficient Semantic Similarity based Feature Selection and Classification Technique for Depression Analysis

P. Suganya[1], Dr. G. Vijaiprabhu[2]

[1]Ph.D Research Scholar, PG and Research Department of Computer Science, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India.
[2]Assistant Professor, PG and Research Department of Computer Science, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India.

Semantic Similarity based Feature Selection introduces a novel approach to enhance sentiment analysis in natural language processing (NLP). As sentiment analysis gains prominence across various domains, the need for effective feature selection becomes paramount to ensure accurate sentiment classification. Traditional methods often overlook semantic nuances present in text, leading to suboptimal performance. To address this, the proposed approach leverages semantic similarity measures to capture the underlying semantics of text more effectively. The chapter explores various semantic similarity metrics, including WordNet-based methods, Word2Vec embeddings, and Smooth Inverse Frequency (SIF), among others, to extract features that better represent the semantic context of the text. The feature selection process involves preprocessing techniques such as tokenization, stop word removal, stemming, and lemmatization to refine the input data. Subsequently, feature extraction methods like WordNet semantic similarity and Word2Vec embeddings are employed to capture semantic relationships within the text. The chapter also delves into feature selection techniques such as TF-IDF, Information Gain, and Gini Index to identify the most informative features for sentiment analysis. Furthermore, ensemble classification methods, including Decision Trees, Ordinary Least Squares Regression (OLSR), Artificial Neural Networks (ANN), and Support Vector Machines (SVM), are utilized to classify sentiments based on the selected features. By integrating these approaches, the proposed methodology aims to improve the accuracy and robustness of sentiment analysis models, thus advancing the state-of-the-art in NLP applications such as opinion mining and social media monitoring. Through experimental evaluation and comparison with traditional methods, the effectiveness of the semantic similarity-based feature selection approach is demonstrated, highlighting its potential to enhance sentiment analysis tasks across diverse datasets and domains.

Keywords: Sentiment analysis, depression, tweet, classification, feature selection, and machine learning.

1. Introduction

In recent years, sentiment analysis has garnered considerable attention in natural language processing (NLP) research due to its broad applicability across domains such as marketing, social media analysis, and customer feedback analysis [1]. Sentiment analysis aims to automatically discern the sentiment expressed in a piece of text, whether it is positive, negative, or neutral. However, the efficacy of sentiment analysis heavily depends on the selection of informative features that effectively capture the underlying semantics of the text [2, 3].

Traditional feature selection methods for sentiment analysis often rely on statistical measures or linguistic patterns to identify relevant features [4]. However, these approaches may overlook the semantic nuances present in the text, potentially leading to suboptimal performance, particularly when dealing with complex and nuanced language [5]. To address this limitation, this research proposes a novel approach for feature selection in sentiment analysis based on semantic similarity [6, 7]. Semantic similarity measures quantify the relatedness between words or phrases based on their semantic meanings, enabling a more nuanced understanding of the text compared to traditional feature selection methods.

In this paper, a comprehensive investigation into the effectiveness of semantic similarity-based feature selection methods for sentiment analysis tasks is presented [8]. Specifically, various semantic similarity metrics, such as Word Embedding-based methods, Semantic Graph-based methods, and Distributional Semantics models, are explored to capture different aspects of semantic relatedness in text data. A framework is proposed that integrates semantic similarity-based feature selection into the sentiment analysis pipeline. The framework includes preprocessing steps for text normalization, feature extraction using semantic similarity metrics, and sentiment classification using machine learning algorithms [9].

To evaluate the proposed approach, experiments are conducted on benchmark sentiment analysis datasets, comparing the performance of the method with traditional feature selection techniques. The impact of different semantic similarity metrics on sentiment analysis accuracy, robustness to domain variations, and computational efficiency is analyzed. This research contributes to advancing the state-of-the-art in sentiment analysis by leveraging semantic similarity-based feature selection methods to improve the accuracy and robustness of sentiment classification models [10]. The findings from this study have implications for various applications requiring accurate sentiment analysis, including opinion mining, social media monitoring, and customer feedback analysis.

2. Related Works

When conducting sentimental analysis on a textual dataset, the words that accurately describe the entire document are found, which is useful when reducing dimensionality, computational time, removing unwanted/ redundant terms, and improving accuracy and performance. The significant terms extracted from the entire document include a description of the content; these words/terms are referred to as keywords. The method used to extract them is referred to as the keyword extraction technique. Monali Bordoloi et al. (2020) [11] presented a co-occurrence graph-based statistical approach to find the global rank of keywords. A new weighting technique is used to improve the standard Node and Edge ranking technique. A keyword can also exhibit bipolarity based on the problem at hand. The author proposed a novel graph-based algorithm that gives a higher priority to the important keyword when compared to the least significant one because the keywords play a prominent role in determining the polarity of the text. Baumgarten et al. (2013) [12] presented a keyword-based sentimental mining approach to analyze the tweets present on Twitter. The authors mainly use the keyword-based classifier to perform sentiment mining in short messages, and the approach can be automatically extended for messages with multiple dimensions. The main challenge encountered here is the non-trivial problem of extracting specific features/aspects from the short messages.

The main aim of the lexicon-based techniques is to identify the sentiments present in documents or sentences by analyzing the sentimental aspects present in the user's writing. The sentiment words are used to express the sentiments such as "happy" (positive)," wow" (positive), "shit" (negative), and "terrible" (negative). These words are known as opinion lexicon or SentiWordNet. Since the lexicon-based approaches use supervised learning, they require an external knowledge source for training. It can be in the form of a labeled lexicon that contains a polarity associated with each sentimental word. The lexicons are also implemented in developing labeled training data for the machine learning classifier. The polarity is mainly expressed using a numerical value that indicates how strong a particular word is associated with a positive and negative polarity.

Even though lexical based techniques are efficient in conducting the sentimental analysis, it often ignores the contextual information associated with the sentence. To overcome this problem, Minghui Huang et al. (2020) developed a lexicon-based attention mechanism for their Sentiment Convolutional Neural Network (SCNN) to analyze both the sentiments and contextual information derived from the sentiment words. The contextual information is mainly captured from the word embeddings, and it is a prominent indicator of sentiments to make effective predictions. This technique offers accuracy, precision, recall, and F1-Measure 88.4%, 88.9%, 88.9%, and 88.9% [13].

Kristína Machovaet al. (2020) applied the lexicon approach for automatic labeling to overcome the complexities associated with ambiguous and subjective manual labeling. To optimize the lexicon labeling approach, Particle Swarm Optimization (PSO) technique is used. The PSO optimizer repeatedly labels every word present in the lexicon and evaluates the opinion classification approach after every optimal label for the words present in the lexicon is identified. This hybrid approach can classify more than 99% of text accurately and achieve better results than the traditional lexicon-based approaches. Mahmoud Al-Ayyoub et al. (2015) presented a lexicon-based sentimental analysis approach for Arabic tweets. The main complexity of the polarity classification is the Arabic language's intricate structure and fewer datasets present for processing. The authors mainly used the lexicon approach to build a large sentimental lexicon and a sentimental analysis engine [14, 15].

Machine learning algorithms serve as useful in handling massive datasets and solving real-world problems. They are classified into two groups, namely supervised and unsupervised learning algorithms. The supervised learning algorithm mainly inputs the labeled dataset, and the accuracy of the outcomes is evaluated using the training dataset. As a consequence, supervised learning is best suited to problems that have a variety of available reference points or ground truth to train the algorithm with Support Vector Machine (SVM), K-Nearest Neighbour (K-NN), Artificial Neural Network, Naïve Bayes, Decision tree, and random forest are some examples of supervised learning. An unsupervised model takes the unlabelled data as an input and extracts the features and patterns present in it without using any external support(manual intervention) and on its own. It resembles a black box structure of processing. A deep learning algorithm is given a dataset with no specific instructions about what to do with it in unsupervised learning. The training dataset is a collection of examples with no particular desired outcome or right answer. The neural network then attempts to automatically find meaning in the data by extracting useful features and analyzing the data's structure. Deep learning techniques mainly use unsupervised learning [16, 17].

3. Semantic Similarity based Feature Selection

This paper applied sentiment analysis-based semantic similarity feature extraction and hybrid feature selection based on the Term Frequency-Inverse Document Frequency (TF-IDF), Information Gain, and Gini Index feature selector for tweet based sentiment detection. Tweet data is filtered using tokenization, stop word removal, stemming, and lemmatization. The feature extraction method uses WordNet, Word2Vector, Smooth Inverse Frequency (SIF), Cosine Similarity (CS), Jensen-Shannon Divergence (JSD), Word Mover's Distance (WMD), Locally Linear Embedding (LLE), and Latent Semantic Indexing (LSI). The different classification methods, such as Certainly, here is the requested sequence: Optimized Link State Routing (OLSR), Artificial Neural Network (ANN), Support Vector Machine (SVM), Random Forest (RF), and Decision Tree (DT), are used to analyze the proposed method performance. The block diagram of the proposed semantic feature extraction (FE) method and the hybrid feature selection (HFS) method is shown in Figure 1.

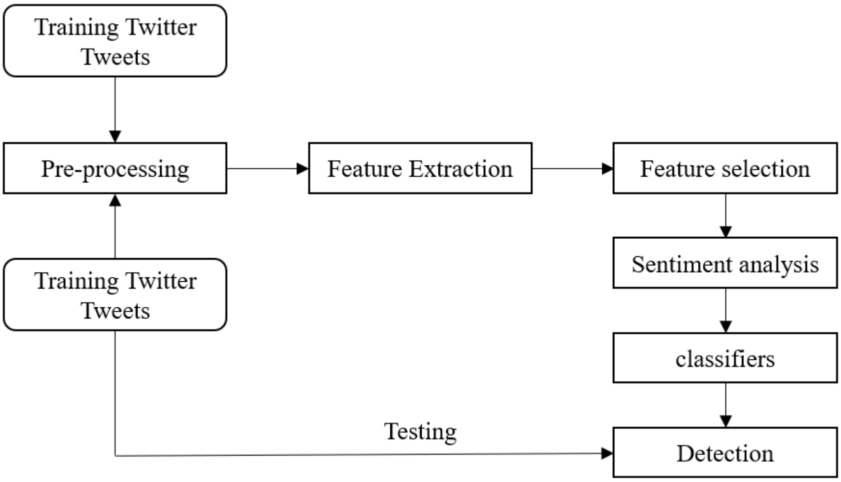

Figure 1. Proposed Methodology

3.1. Pre-processing Techniques

Preprocessing techniques are commonly used in the NLP methods to reduce the redundant information from the dataset. Removing unwanted information in tweet data; helps to improve classification performance. The preprocessing techniques used in this research are tokenization, stop word removal, stemming, and lemmatization.

Tokenization

The tokenization method is involved in separates the composite text in the datasets into small tokens. The proposed model applies an N-gram tokenizer to eliminate delimiters and word spaces in the composite text.

Stop Word Removal

Stop words denote the most common words in a language, such as of is the and at. Most researches in NLP consider stopping words, which affects the model's performance and is removed from the input data before the feature extraction and selection process. The pre-compiled lists are the standard method to remove stop words from the input data, and it is used in this research.

Stemming and Lemmatization

Stemming and lemmatization are commonly used in NLP models to give a normalized form of input data. Stemming performs a basic form of approximation and does not replace the word. The lemmatization method completely removes or replaces the suffix completely from input data to form a lemma.

3.2. Feature Extraction

Feature Extraction (FE) methods such as WordNet, Word2Vector, Smooth In- verse Frequency, Cosine Similarity, Jensen Shannon Distance, Word Mover Distance, Local Linear Embedding and Latent Semantic Index were used.

WordNet Semantic Similarity

Based on a set of terms that describe each term's properties, WordNet semantic similarity measures the similarity between two terms. WordNet uses the relationship with other similar terms in the hierarchical structure data. WordNet considers terms characteristics to measure similarity between different concepts, ignoring position and information on the taxonomy. WordNet can be calculated using the Equation (1).

$$\mathrm{Sim_t}(\mathrm{C1},\mathrm{C2}) = \frac{|\mathrm{C1} \cap \mathrm{C2}|}{|\mathrm{C1} \cap \mathrm{C2}| + \alpha|\mathrm{C1} \cap \mathrm{C2}| + (\alpha-1)|\mathrm{C2}-\mathrm{C1}|} \text{--------}(1)$$

In the context of similarity identification between C1 and C2 concepts with multiple senses, the metric result is given by the maximum between the values of the similarity metric of each sense of concepts C1and C2. For that, noting the general similarity measure with $\mathrm{Sim_t vsk}(\mathrm{C1},\mathrm{C2})$. The uncommon characteristics of relative importance are denoted as a belongs to [0, 1]. The value of α increases with the similarity of terms and decreases the difference between the terms. The determination of α is based on observation and not necessarily a symmetric relation.

Word2 Vector

Word vectorization, is a natural language processing (NLP) process. The process uses language models to map words into vector space. A vector space represents each word by a vector of real numbers. It also allows words with similar meanings have similar representations. Word2Vec is used to express words based on the vector representation, as shown in Equation (2).

$\mathrm{V} = (\mathrm{v}_1, \mathrm{v}_2, \mathrm{v}_3 \ldots . \mathrm{v}_\mathrm{n})$---------(2)

Where word space is denoted as, V and vector of word space is denoted as V1, V2 of particular data.

Smooth Inverse Frequency

Smooth Inverse Frequency (SIF) is the sentence embedding method and is highly used in NLP due to its simplicity and competitive performance. Consider the context vector $\mathrm{C}\varepsilon\mathrm{R}^\mathrm{d}$, the word, w probability is emitted in the context, using Equation (3).

$\mathrm{P}(\mathrm{w}\backslash\mathrm{c}) = \mathrm{p}(\mathrm{w}) + (1 -) \left(\frac{\exp(\mathrm{w},\mathrm{c})}{\mathrm{z}_\mathrm{c}}\right)$---------(3)

Where a scalar hyper parameter is denoted as aß ∈ [0, 1], word embedding for ω is denoted as ω ∈ R^d, common discourse is denoted as c_o ∈ R^d, and the normalizing constant is represented as Z_c = $\sum_{\omega \in \mathrm{W}} \exp\,(\mathrm{c}, \mathrm{w})$.

Cosine Similarity

Cosine Similarity (CS) is easy to interpret and simple to compute for sparse vector matrices, as it is widely used in information retrieval and text mining methods. Cosine similarity measures the cosine angle between two vectors, as shown in Equation (4). The document with different totals of the same composition is allowed to be treated identically, making this method popular for text analysis.

$\mathrm{S}_{\mathrm{cosine}}(\mathrm{x}, \mathrm{y}) = \frac{\mathrm{x}'\mathrm{y}}{||\mathrm{x}||||\mathrm{y}||}$---------(4)

Where ||x||=$\sqrt{\sum_{\mathrm{i}=1} \mathrm{x}_1 2}$and ||y||$\sqrt{\sum_{\mathrm{i}=1} \mathrm{y}_1 2}$are the lengths of the vector x and y respectively.

Jensen Shannon Distance

The Jensen Shannon (JS) distance is based on the KullbackLeibler (KL) distance, which is also indexed to measure the similarity of two probability distributions that help solve the asymmetry problem. The formula for JS is showed in Equation 5.

$\mathrm{JS}(\mathrm{P}||\mathrm{Q}) = \frac{1}{2}\mathrm{KL}(\mathrm{P}| \left|\frac{\mathrm{P}+\mathrm{Q}}{2}\right) + \frac{1}{2}\mathrm{KL}(\mathrm{Q}| \left|\frac{\mathrm{P}+\mathrm{Q}}{2}\right)$---------(5)

The KL divergence can be calculated as the negative sum of the probability of each event in P multiplied by the log of the probability of the event in Qover the probability of the event in P. The JS value presents between 0 to 2. The KL denotes $\mathrm{D}(\mathrm{P}||\mathrm{Q})$ and formula of KL is shown in Equation (6).

$\mathrm{D}(\mathrm{P}||\mathrm{Q}) = \sum_{\mathrm{x} \in \mathrm{x}} \mathrm{P}(\mathrm{X}) \log \left(\frac{\mathrm{P}(\mathrm{x})}{\mathrm{Q}(\mathrm{x})}\right)$---------(6)

Word Movers Distance

The Word Movers Distance (WMD) was introduced based on the earth movers distance, which provides a solution to transportation problems. WMD measures the distance between two text documents x, y ∈ X considering the distance between the words. The number of distinct words in x and y is denoted as $|x|, |y|$. The normalized frequency vectors of each word in the documents x and y is denoted as $f_x \in R^{|x|}, f_y \in R^{|y|}$, respectively. The WMD distance between the two documents x and y is defined in Equation (7-8).

$\text{W M D}\,(x, y) = \min_{F \epsilon R|x|*|y|(C,F)}$---------(7)

$s.t., F1 = f_x F^T 1 = f_y$---------(8)

Where transportation flow matrix is denoted as F, F_{ij}, represent the flow travelling amount from i^{th} word, x_i in x to j^{th} word y_i in y, and transportation cost is denoted as C as C_{ij}= dist (v_{xi}, v_{yi}) is distance between two words evaluated in the Word2Vec embedding space. The euclidean distance dist (v_{xi}, v_{yi}) = $|v_{xj}, v_{yj}|$ is popular choice and used in this research.

Local Linear Embedding

Linear locally embedding represents each data point; based on a linear combination of k nearest neighbours. The LLE can be expressed as follows in Equation (9).

$\min_w m \sum_{i=1} n \backslash x_i^m - \sum_{j=1} k W_{IJ}^M x_{iJ}^m |2, \sum_J k W_{IJ}^M = 1$ ---------(9)

The number of neighbours is denoted as k and a linear relationship weighting factor is denoted as W_{ij}^m. The neighbourhood sample x_i doesnt have sample x_j and this is set as$W_{ij} = 0$. The Lagrange multiplier method is denoted in Equation (9) and LLE is measured based on formula in Equation (10-11).

$\min_F \sum_{m=1} 2\alpha_m \text{tr}\,(FA_m F^T)$---------(10)

$A_m = (I - W_m)^T (I - W_m)$---------(11)

The LLE common subspace is denoted as F and matrix $W_m$with W_{ij}^m elements, m^{th} modal is represent as, m to represent the text modality. The mani fold structure controlling parameter is denoted as am to preserve m^{th} modality item.

Latent Semantic Index (LSI)

The LSI method is developed for a text retrieval method. The LSI method measures Singular Value Decomposition (SVD) on the term-document matrix. A new matrix is constructed to provide the original term-document matrix based on first maximal T singular values and respective singular vectors. The dimension of the new matrix is reduced by removing noise which helps to achieve excellent retrieval performance. The term document matrix is denoted as $A_{K \times N}$ Equation (12) that related to K terms in N documents. Based on SVD, the

Matrix $A_{K \times N}$ is split into three matrices, such as.

$A_{K \times N} = U_{K \times N} S_{n \times n} (V_{N \times n})$---------(12)

Where the number of documents is denoted as N, the number of terms is denoted as K, n = min (K, N), U and V have orthogonal columns, i.e. UU^T=$V^T V$=1,the singular values of $A_{K \times N}$, and the singular values are sorted in non-increasing order so that $\delta_i \geq \delta_i$, fori, j. The truncated SVD of $A_{K \times N}$ is selected based on the first maximum of T singular values from matrix S and keeping the corresponding columns in U and V, as given in Equation (13).

$A_{K\times N} = U_{K\times T}S_{T\times T}(V_{N\times T})$---------(13)

In the least squares sense, the $A_{K\times N}$ is the best approximation to $A_{K\times N}$of any rank-T. The matrix $A_{K\times N}$ can be denoted in reduced dimension and latent semantic feature space is given in Equation (14).

$A_{T\times N} = S_{T\times T}(V_{N\times T})$---------(14)

Where latent space dimensionality is denoted as T, and each column of $A_{K\times N}$ corresponds to a latent semantic feature of each training dataset. The normalized projection features are denoted as W (B) = [W (u_1, B)] ....., W (u_k, B) and its latent semantic feature is denoted as in Equation (15).

$\emptyset(B) = (U_{K\times T})W(B)$---------(15)

3.3. Feature Selection

Feature selection is the process of reducing the number of input variables when developing a predictive model. Feature Selection (FS) methods such as TF-IDF,Information Gain (IG) and Gini Index (GI) were used in this research for tweet spam/ham detection.

Term Frequency-Inverse Document Frequency

TF-IDF is developed from IDF with the heuristic intuition term occurs less frequently in the document that is a good discriminator and should be given more weight for the term. The TF-IDF term weighting formula is given in Equation (16).

$W_{i,j} = t\,f_{i,j} \times \log\left(\frac{N}{df_i}\right)$---------(16)

Where term weight is denoted as$W_{i,j}$ for term i in document j, the collected number of documents is denoted as N, the term frequency is denoted as $t\,f_{i,j}$, and the document frequency is denoted as df_i.

Information Gain

The Information Gain (IG) is used in gene analysis that can be used to evaluate difference between conditional entropy. The IG reduction of uncertainty is denoted as g (X, Y), as shown in Equation (17).

$g(Y, X) = H(Y) - H(Y|X)$---------(17)

Where Y dataset entropy is denoted as H (Y) that measures the uncertainty in- volved in predicting random variable value. The conditional entropy is denoted as H (Y|X) that represent known variable X uncertainty. The probability distribution is denoted as H (Y) and H (Y|X) can be measured in Equation (18- 19).

$H(Y) = -\sum p(y)\log p(y)$---------(18)

$H(Y\backslash X) = -\sum_{x\in X} p(x)H(Y\backslash X = x)$---------(19)

Gini Index

Data samples are denoted as S, the various class label attribute that denotes various classes of C_i (i = 1, 2, 3, ...m). Based on the class labels attribute values, S can be divided into m subsets

S_i (i = 1, 2, 3, ...m). If subset samples S_i belongs to class C_i and the number of samples in the subset is S_i, the gini index is denoted as in Equation (20).

$\text{GiniIndex}\,(S) = 1 - \sum_{i=1} mP_i^2$---------(20)

Where probability $P_i$of any sample C_i estimate by S_i. The gini index initial form is used to measure impurity attribute for classification. The gini index equation is shown in Equation (21).

$\text{GiniIndex}\,(S) = \sum_{i=1} mP_i^2$---------(21)

3.4. Ensemble Classification

The classifier uses the selected features from the FS method to classify the sentiment of the tweet. The classifiers detect spam tweets from the input data based on sentiment analysis and selected features.

Decision Tree (DT)

The decision tree model, a series of simple rules, is applied to segment the data that are denoted in the empirical tree. The rules perform the repetitive pro-cess of splitting the data for segmentation. The C5.0 is an improved version of C4.5 that differs as follows: (i) a nominal split has a default branch-merging option; (ii) misclassification costs can be denoted; (iii) crossvalidation and boosting are available (iv) the ruleset algorithm is improved. The DT model has lower efficiency than neural networks for nonlinear data and is also affected by noisy data.The model is more suitable to predict categorical outcomes if sequential patterns and visible trends are available. The decision tree model has lower efficiency in time-series analysis.

Ordinary Least Squares Regression(OLSR)

An OLSR is a linear approximation that reduces the sum of the squares of the distances between the observation points and the estimated points. The slope formula of ordinary least squares estimation is B. Ordinary least squares is more suitable for the cases in which one of the two variables in Equation (22).

$\beta = (X^T X)^{-1} X^T y$ ---------(22)

Where the matrix regressor variable X, T matrix transpose and y vector of the value of the response variable.

Artificial Neural Network

Artificial Neural Network (ANN) is a popular Machine Learning (ML) method proliferating in recent years. ANN model can handle non-linear data and provide adequate performance; developed a multilayer neural architecture is a computation model. The human nervous system inspires the ANN, and the learning process of the ANN is based on pattern analysis of the network. ANN method is based on two processes, namely forward process and backpropagation. In the activated network layer of the forward process, the signals are processed in the forward direction, i.e., input to output. The error correction is the backward process based on bias term and connection weight. The backpropagation applies a gradient descent rule at each learning cycle to minimize the network error. This method is repeated until the desired result is achieved with many references related to neural networks with the neural net model. The error value is used to weigh the outputs and summed up in the output neuron. The input and output layer is explained as follows.

- Input Unit: 0_1^1 = y
- Hidden Units: 0_i^2 = f (net_i), i = 1...., I
- Net, $i = y \times W_i^1 + b_i$, where f is the sigmoid activation function
- Output Unit: N (Y) = $\sum_{i=1} L(W_i^2, 0_i^2) = \sum_{i=1} L(W_i^1, f(W_i^1, W_i^1, y + b_i))$

Support Vector Machine

The support vector machine is based on the statistical learning method that uses the hyperplane to classify the data into various categories. The hyperplane is developed from a given dataset. The training feature dataset instances are labelled as {(x, y)},i = 1,2,3,...N, where the number of instances is denoted as N, y_i is the class of instance 2, from input data. In an SVM, the maximum margin separating the hyperplane is developed based on the closest points in high dimensional space. SVM computes the sum of distances between the hyper plane points to close points in high dimensional space to evaluate margin. The margin boundary function is computed as in Equation (23).

$$\text{Minimise} = W(\alpha) = \frac{1}{2}\sum_{i=1} N \sum_{j=1} N y_i, y_j, \alpha_i, \alpha_j, K(x_i, x_j) - \sum_{i=1} N\alpha_i \quad \text{---------(23)}$$

Where α is a vector of N variables and soft margin parameter is denoted as C, C>0. The SVM kernel function is denoted as k (x_i, x_j). In this research, Radial Basis Function (RBF) kernel is used, as in Equation (24).

$$k(x_i, x_j) = \exp(-\gamma|x_i - x_j|^2), \gamma \text{ ---------(24)}$$

Where γ, r and d are kernel parameters.

Ensemble Learning

The ensemble of classifiers employed for sentiment analysis and spam tweet detection encompasses Decision Tree (DT), Ordinary Least Squares Regression (OLSR), Artificial Neural Network (ANN), and Support Vector Machine (SVM). Decision trees, represented as a series of simple rules, partition data based on empirical tree structures, with enhancements such as those found in the C5.0 model, offering improved features like nominal split options and cross-validation. While decision trees excel in identifying categorical outcomes with discernible trends, they may exhibit lower efficiency with non-linear data and susceptibility to noise. OLSR, a linear approximation method, minimizes the sum of squared distances between observed and estimated points, making it suitable for linear relationships in data. ANN, inspired by the human nervous system, leverages multi-layer architectures to handle nonlinear data and undergoes learning via forward and backward processes, utilizing backpropagation for error correction. SVM, a statistical learning method, employs hyperplanes to segregate data into distinct categories, optimizing margins through kernel functions like the Radial Basis Function (RBF). By combining the unique strengths of these classifiers, the ensemble approach aims to enhance the accuracy and robustness of sentiment analysis and spam tweet detection tasks, accommodating diverse data characteristics and improving overall performance.

4. Result and Discussion

The dataset utilized in this study is the sentiment140 dataset, comprising a vast collection of 1,600,000 tweets sourced through the Twitter API. These tweets have undergone manual annotation to assign sentiment polarity labels, where a label of 0 signifies negative sentiment and 4 represents positive sentiment, rendering it conducive for sentiment analysis tasks. Within the dataset, six key fields are present for each tweet entry. Firstly, the "target" field indicates the polarity of the tweet, with values ranging from 0 to 4, encompassing negative, neutral, and positive sentiments. Secondly, the "ids" field assigns a unique identifier to each tweet for tracking and organizational purposes. The "date" field captures the timestamp denoting the date and time of tweet creation, adhering to the UTC (Coordinated Universal Time) standard. Furthermore, the "flag" field records any associated query term if the tweet originates from a specific query; otherwise, it is marked as NO_QUERY. The "user" field contains the username of the Twitter account responsible for the tweet, while the "text" field encapsulates the actual textual content of the tweet, encompassing hashtags, mentions, emoticons, and other textual features [18]. This dataset serves as a valuable resource for developing and evaluating sentiment analysis algorithms and models, offering a diverse range of annotated tweets spanning various sentiments.

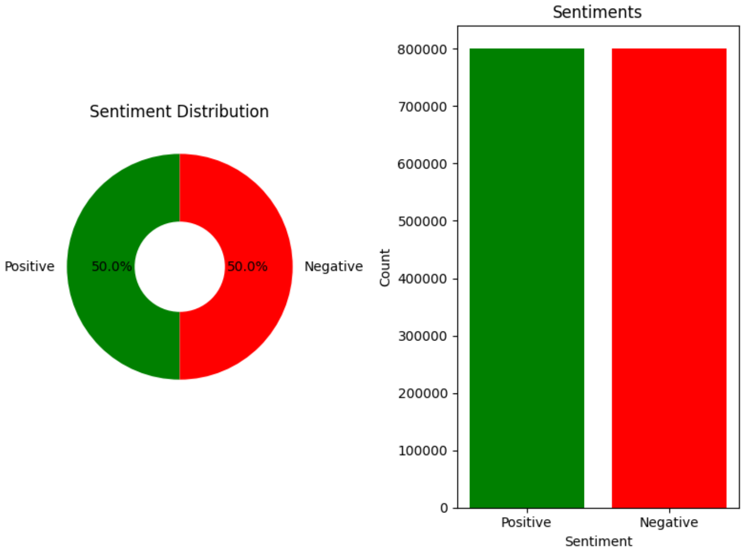

Figure 2. Distribution of Sentiment

This figure provides an overview of the distribution of sentiment within the dataset, focusing specifically on positive and negative sentiments. It likely depicts the proportion or frequency of tweets categorized as positive and negative, visually representing the balance between these two sentiment polarities. Understanding this distribution is crucial for assessing the prevalence of positive and negative sentiments in the dataset and for informing subsequent sentiment analysis tasks.

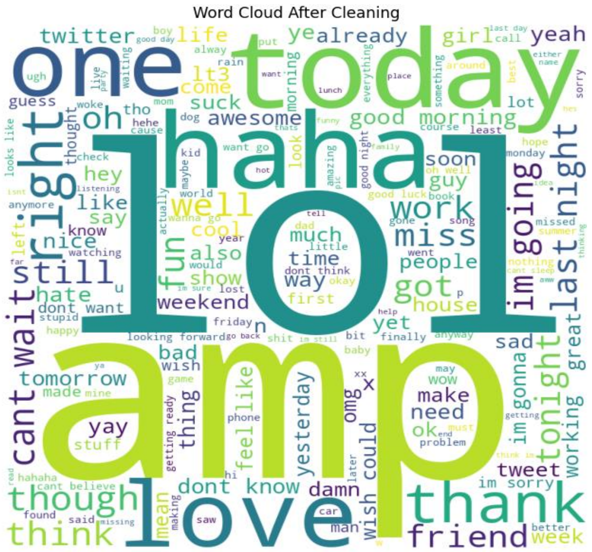

Figure 3. Pre-Processed Word Cloud

This figure illustrates a word cloud generated from the pre-processed text data, emphasizing words associated with positive and negative sentiments. By visualizing the most frequent terms after preprocessing, the word cloud offers insights into the prevalent themes and sentiments expressed in the dataset. Positive and negative words are likely highlighted with different colors or font sizes, enabling quick identification of key features contributing to sentiment classification.

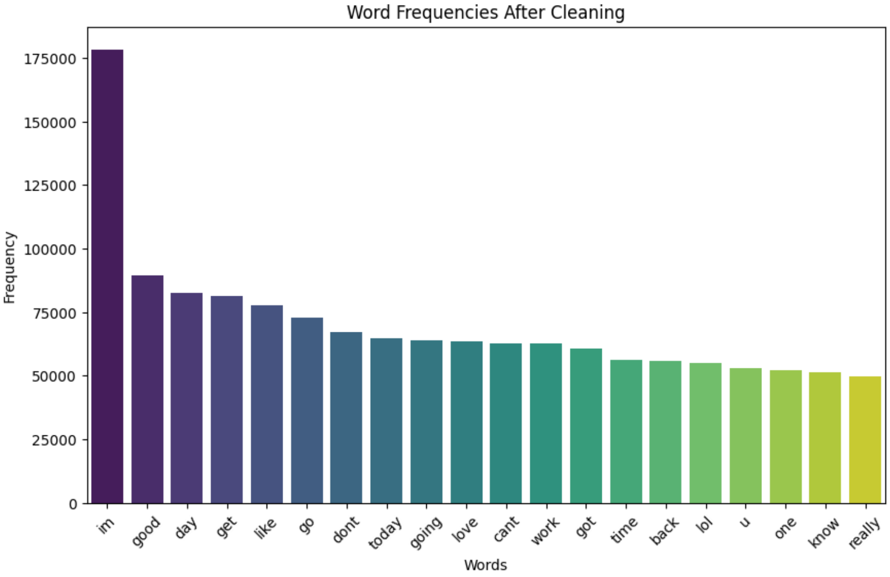

Figure 4. Histogram of Words

This figure presents a histogram focusing on the frequency distribution of words related to positive and negative sentiments. By plotting the frequency of positive and negative words separately, this visualization enables researchers to identify the most common terms associated with each sentiment polarity. Analyzing the histogram helps uncover important keywords or linguistic patterns that characterize positive and negative sentiments in the dataset, facilitating more accurate sentiment analysis and interpretation.

The proposed semantic-based similarity feature extraction method measured the evaluation metrics such as Accuracy, Precision, Recall, and RMSE values. Theformula for Accuracy, Precision, Recall, and RMSE is as in equation (25 -28).

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

$$precision = \frac{TP}{TP + FP}$$

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

$$RMSE = \sqrt{\sum_{i=1} n \left(\frac{y_{i-}y_j}{n}\right)^2}$$

Table 1. Comparison of Accuracy

Feature Selection Methods	OLSR	DT	ANN	SVM	EL
WordNet	73	81	88	86.34	88.45
Word2Vector	74	83.45	88.45	86.92	89.99
Smooth Inverse Frequency (SIF)	75.34	84.1	89	87.2	90.2
Cosine Similarity (CS)	76.54	84.9	89.56	88	91.56
Jensen-Shannon Divergence (JSD)	78	85.3	90.76	89.56	92
Word Mover's Distance (WMD)	79.4	86.09	91	90.45	92.67
Locally Linear Embedding (LLE)	80	87	91.23	91	93.1
Latent Semantic Indexing (LSI)	80.55	88	92	91.56	94

Table 1 provides a comprehensive comparison of accuracy scores achieved by different feature selection methods across various classifiers, including Ordinary Least Squares Regression (OLSR), Decision Tree (DT), Artificial Neural Network (ANN), Support Vector Machine (SVM), and Ensemble Learning (EL). Notably, as we move from traditional methods like WordNet to more advanced techniques such as Latent Semantic Indexing (LSI), there is a consistent trend of improvement in accuracy across all classifiers. For instance, the accuracy scores increase from 73% with WordNet to 94% with LSI for the EL classifier, indicating the effectiveness of utilizing semantic similarity-based feature selection methods. In terms of percentage difference, we observe varying degrees of improvement, with the most substantial improvements seen when transitioning from WordNet to more advanced methods like LSI, where the percentage difference ranges from 0.55% to 2.25%.

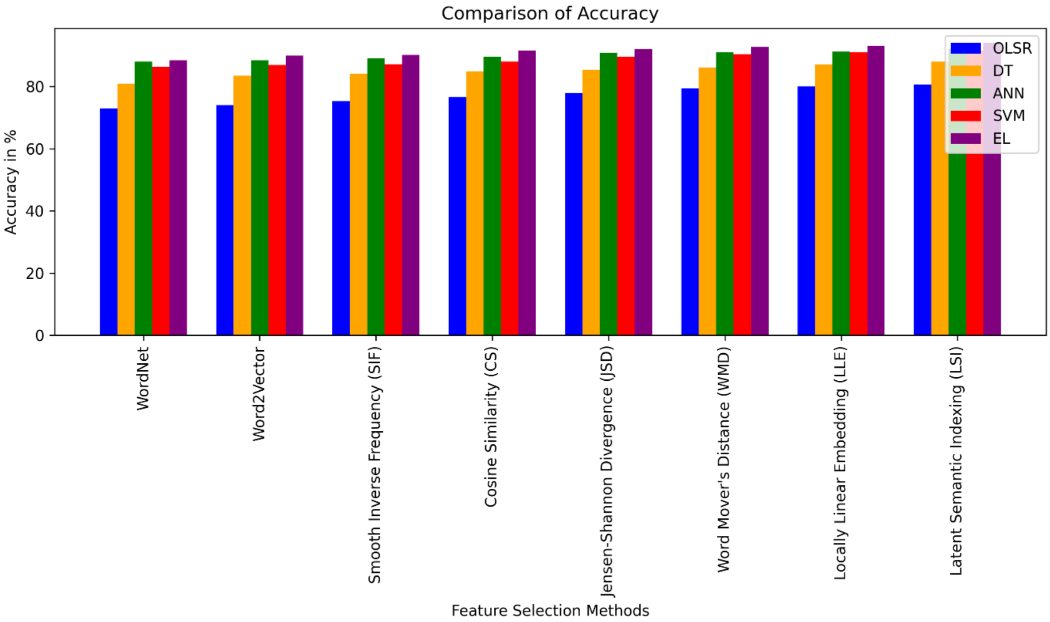

Figure 5. Comparison of Accuracy

Table 2. Comparison of Precision

Feature Selection Methods	OLSR	DT	ANN	SVM	EL
WordNet	70	77	84	81.14	84.35
Word2Vector	71.1	80.45	85	81.92	84.69
Smooth Inverse Frequency (SIF)	75.34	81.3	86.2	82.3	85.2
Cosine Similarity (CS)	75.54	81.49	86.3	82.7	86.56
Jensen-Shannon Divergence (JSD)	76.34	82.3	87.2	83	87
Word Mover's Distance (WMD)	77.2	82.9	88	83.4	88.67
Locally Linear Embedding (LLE)	78.3	83	88.56	83.9	89.1
Latent Semantic Indexing (LSI)	80.55	88	92	91.56	92

Table 2 presents a comparison of precision scores across the same feature selection methods and classifiers. We observe a similar trend of increasing precision scores with more sophisticated feature selection techniques. This suggests that advanced feature selection methods like Cosine Similarity (CS) and Word Mover's Distance (WMD) contribute to higher precision in sentiment analysis tasks across various classifiers. The percentage difference analysis further highlights the relative improvements achieved with each feature selection method, indicating the effectiveness of more advanced techniques in enhancing precision.

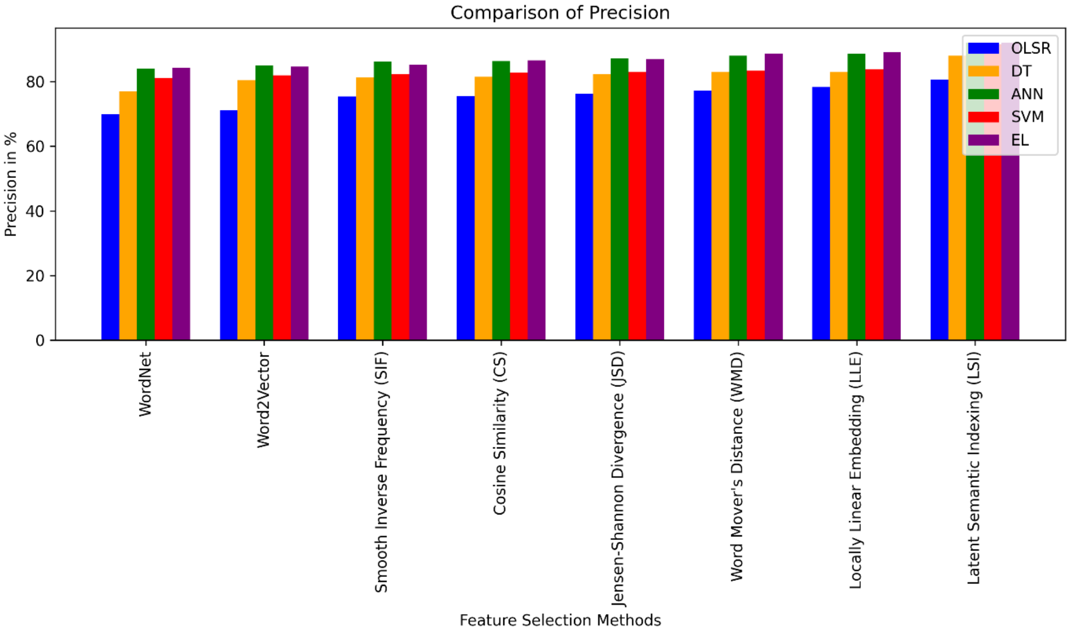

Figure 6. Comparison of Precision

Table 3. Comparison of Recall

Feature Selection Methods	OLSR	DT	ANN	SVM	EL
WordNet	70.8	76.4	82.3	83.14	85.35
Word2Vector	71.9	80	82.5	84.3	85
Smooth Inverse Frequency (SIF)	75	81	83	85	86.2
Cosine Similarity (CS)	75.3	81.7	83.6	86.3	87.56
Jensen-Shannon Divergence (JSD)	76	83	84	86.9	87.65
Word Mover's Distance (WMD)	77	83.4	84.8	87	88
Locally Linear Embedding (LLE)	78	83.8	85.8	87.8	89.16
Latent Semantic Indexing (LSI)	79	84	89	88	93

Table 3 compares the recall scores achieved by different feature selection methods and classifiers. Once again, we notice a pattern of improved performance with advanced feature selection techniques, with LSI consistently outperforming other methods across classifiers. This indicates that LSI effectively captures semantic relationships in the text data, leading to better recall rates for sentiment analysis. The percentage difference analysis underscores the relative improvements in recall achieved with each feature selection method, with the transition to advanced methods resulting in significant gains in performance. The tables and percentage difference analysis provide valuable insights into the impact of different feature selection methods on the performance of sentiment analysis classifiers, highlighting the effectiveness of advanced techniques in improving accuracy, precision, and recall.

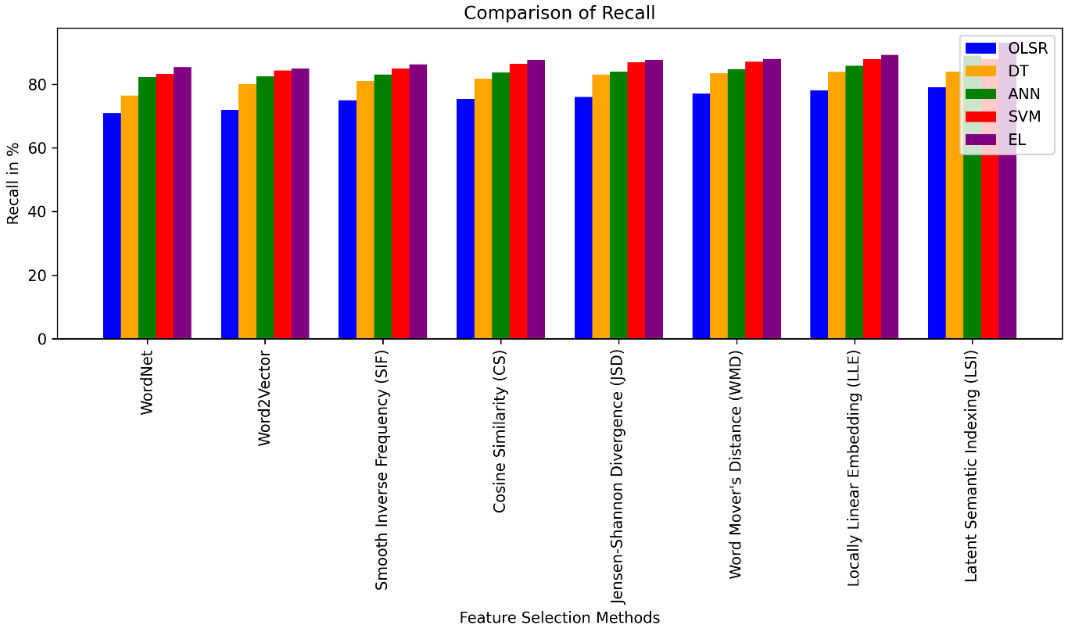

Figure 7. Comparison of Recall

Table 4. Comparison of RMSE

Feature Selection Methods	OLSR	DT	ANN	SVM	EL
WordNet	0.567	0.546	0.511	0.482	0.421
Word2Vector	0.597	0.512	0.490	0.475	0.412
Smooth Inverse Frequency (SIF)	0.651	0.623	0.531	0.463	0.396
Cosine Similarity (CS)	0.465	0.432	0.542	0.423	0.367
Jensen-Shannon Divergence (JSD)	0.492	0.421	0.521	0.423	0.341
Word Mover's Distance (WMD)	0.621	0.541	0.511	0.413	0.332
Locally Linear Embedding (LLE)	0.633	0.582	0.499	0.399	0.312
Latent Semantic Indexing (LSI)	0.641	0.591	0.482	0.389	0.293

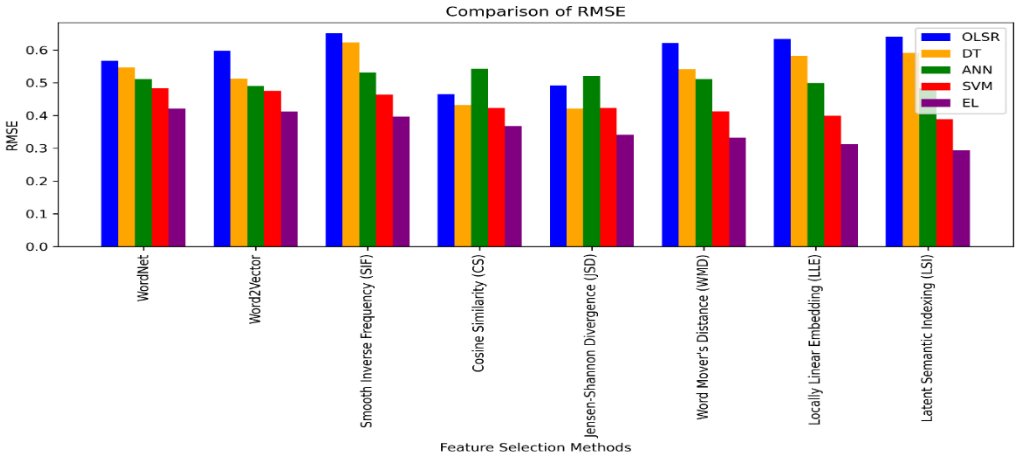

Figure 8. Comparison of RMSE

5. Conclusion

In this paper, the sentiment analysis-based semantic feature extraction and the hybrid feature selection method were used to increase the efficiency of negativity detection in tweets. This research involves applying sentiment analysis as one of the features, along with semantic feature extraction and hybrid feature selection methods. The sentiment analysis measures the polarity of input tweets to improve the efficiency of spam classification. The proposed approach employs the WordNet ontology and applies different semantic based methods and similarity measures for reducing the huge number of extracted textual features, and hence the space and time complexities are reduced. The next paper discusses tweets-based sentiment classification using optimization methods.

References

1. Priyambodo, T. K., Wijayanto, D., & Gitakarma, M. S. (2020). Performance optimization of MANET networks through routing protocol analysis. Computers, 10(1), 2.
2. Veeraiah, N., Khalaf, O. I., Prasad, C. V. P. R., Alotaibi, Y., Alsufyani, A., Alghamdi, S. A., &Alsufyani, N. (2021). Trust aware secure energy efficient hybrid protocol for manet. IEEE Access, 9, 120996-121005.
3. Tharini, V. J., & Shivakumar, B. L. (2022). High-utility itemset mining: fundamentals, properties, techniques and research scope. In Computational intelligence and data sciences (pp. 195-210). CRC Press.
4. Kurode, E., Vora, N., Patil, S., & Attar, V. (2021, August). MANET routing protocols with emphasis on zone routing protocol–an overview. In 2021 IEEE Region 10 Symposium (TENSYMP) (pp. 1-6). IEEE.
5. Soni, G., Jhariya, M. K., Chandravanshi, K., & Tomar, D. (2020, February). A multipath location based hybrid DMR protocol in MANET. In 2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE) (pp. 191-196). IEEE.
6. Chen, Z., Zhou, W., Wu, S., & Cheng, L. (2020). An adaptive on-demand multipath routing protocol with QoS support for high-speed MANET. IEEE Access, 8, 44760-44773.
7. Tharini, V. J., & Shivakumar, B. L. (2023). An efficient pruned matrix aided utility tree for high utility itemset mining from transactional database. International Journal of Intelligent Systems and Applications in Engineering, 11(4s), 46-55.
8. Vu, Q., Hoai, N., & Manh, L. (2020). A survey of state-of-the-art energy efficiency routing protocols for MANET.
9. Kurniawan, A., Kristalina, P., & Hadi, M. Z. S. (2020, September). Performance analysis of routing protocols AODV, OLSR and DSDV on MANET using NS3. In 2020 international electronics symposium (IES) (pp. 199-206). IEEE.
10. AlKhatieb, A., Felemban, E., & Naseer, A. (2020, April). Performance evaluation of ad-hoc routing protocols in (FANETs). In 2020 IEEE wireless communications and networking conference workshops (WCNCW) (pp. 1-6). IEEE.
11. Bordoloi, M., Chatterjee, P. C., Biswas, S. K., & Purkayastha, B. (2020). Keyword extraction using supervised cumulative TextRank. Multimedia Tools and Applications, 79(41), 31467-31496.
12. Baumgarten, M., Mulvenna, M. D., Rooney, N., & Reid, J. (2013). Keyword-based sentiment mining using twitter. International Journal of Ambient Computing and Intelligence (IJACI), 5(2), 56-69.
13. Huang, M., Xie, H., Rao, Y., Liu, Y., Poon, L. K., & Wang, F. L. (2020). Lexicon-based

sentiment convolutional neural networks for online review analysis. IEEE Transactions on Affective Computing, 13(3), 1337-1348.

14. Machová, K., Mikula, M., Gao, X., & Mach, M. (2020). Lexicon-based sentiment analysis using the particle swarm optimization. Electronics, 9(8), 1317.
15. Al-Ayyoub, M., Essa, S. B., & Alsmadi, I. (2015). Lexicon-based sentiment analysis of Arabic tweets. International Journal of Social Network Mining, 2(2), 101-114.
16. Ramadhani, A. M., & Goo, H. S. (2017, August). Twitter sentiment analysis using deep learning methods. In 2017 7th International annual engineering seminar (InAES) (pp. 1-4). IEEE.
17. Neethu, M. S., & Rajasree, R. (2013, July). Sentiment analysis in twitter using machine learning techniques. In 2013 fourth international conference on computing, communications and networking technologies (ICCCNT) (pp. 1-5). IEEE.
18. https://www.kaggle.com/general/35739