Salud, Ciencia y Tecnología - Serie de Conferencias. 2024; 3:702
doi: 10.56294/sctconf2024702

Categoría: STEM (Science, Technology, Engineering and Mathematics)

ORIGINAL

Sentence level Classification through machine learning with effective feature extraction using deep learning

Clasificación a nivel de oración mediante aprendizaje automático con extracción efectiva de características mediante aprendizaje profundo

Savitha D[1], Sudha L[2]

[1]Research Scholar, Asst. Professor of Information Technology, Vellalar College for Women (Autonomous), Erode – 12.
[2]Research Supervisor, Associate Professor, School of Computer Science, VET Institute of Arts and Science College, Erode – 12.

Citar como: D S, L S. Sentence level Classification through machine learning with effective feature extraction using deep learning. Salud, Ciencia y Tecnología - Serie de Conferencias. 2024; 3:702. https://doi.org/10.56294/sctconf2024702

Submitted: 06-12-2023 **Revised:** 24-02-2024 **Accepted:** 16-04-2024 **Published:** 17-04-2024

Editor: Dr. William Castillo-González

ABSTRACT

Social networking website usage has increased dramatically during the past few years. Users can read other users' views, which are categorized into several sentiment classes on this medium with an array of data. These opinions are becoming more and more important while making decisions. To address the above-mentioned issues and improve the sentence-level classification's classification rate, this work introduces a new extensive pinball loss function based twin support vector machine with Deep Learning the (EPLF-TSVM-DL) to identify the polarity (negative and positive) of sentiment sentences. There are four primary components of this technique: The first portion consists of pre-processing the data to minimize noise and improve quality; the second part utilizes word embedding techniques to transform textual data into numerical data. The third part is the CNN for an efficient automatic method of extracting the features-based feature extraction and final is EPLF-TSVM-DL is used for sentence level classification that forms two classes such as Negative and Positive. The findings demonstrated that the EPLF-TSVM-DL outperforms the other classifiers with respect to of time consumption, convergence, complexity, and stability as well as true negative, true positive, error rate, false positive, precision, false negative, and classification rate.

Keywords: Social Networking; Sentiment Classes; Sentence-Level Classification; Data-Pre-Processing; Deep Learning.

RESUMEN

El uso de sitios web de redes sociales ha aumentado dramáticamente durante los últimos años. Los usuarios pueden leer las opiniones de otros usuarios, que se clasifican en varias clases de sentimientos en este medio con una serie de datos. Estas opiniones son cada vez más importantes a la hora de tomar decisiones. Para abordar los problemas mencionados anteriormente y mejorar la tasa de clasificación de la clasificación a nivel de oración, este trabajo presenta una nueva máquina de vectores de soporte gemelo basada en función de pérdida de pinball extensa con aprendizaje profundo (EPLF-TSVM-DL) para identificar la polaridad (negativa y positiva). De oraciones de sentimiento. Hay cuatro componentes principales de esta técnica: la primera parte consiste en preprocesar los datos para minimizar el ruido y mejorar la calidad; la segunda parte utiliza técnicas de incrustación de palabras para transformar datos textuales en datos numéricos. La tercera parte es CNN para un método automático eficiente de extracción de características basada en características y la final es EPLF-TSVM-DL que se utiliza para la clasificación a nivel de oración que forma dos clases, como Negativa y Positiva.

Los hallazgos demostraron que el EPLF-TSVM-DL supera a los otros clasificadores con respecto al consumo de tiempo, la convergencia, la complejidad y la estabilidad, así como en verdadero negativo, verdadero positivo, tasa de error, falso positivo, precisión, falso negativo y tasa de clasificación.

Palabras clave: Redes Sociales; Clases de Sentimientos; Clasificación a Nivel de Oraciones; Preprocesamiento de Datos; Aprendizaje Profundo.

INTRODUCTION

The research of people's ideas, feelings, assessments, attitudes, and emotions regarding entities and their characteristics as they are conveyed in written form is known as sentiment analysis (SA).[1] As social media on the internet, including reviews, blogs, and comments, a growing number of people are expressing their thoughts online. Consequently, the significance of this intriguing issue in commerce and society is growing. Sentence-level sentiment analysis is the primary areas of interest for sentiment analysis. The majority of the previous study on this subject concentrated on utilizing linguistic cues taken from sentences' textual content to determine a sentence's polarity.[2,3] Without taking into account various phrase forms, they approached this task as a universal problem and solved it. Still, many sentence forms convey emotion in rather distinct ways.

Sentiment writing, as opposed to factual language, often communicates in a more random or subtle way, which renders it challenging to identify by only examining each word on its own. One method is unlikely to fit all problems, according to certain arguments.[4] distinct sorts of sentences may require distinct treatments on sentence-level sentiment analysis, meaning that a divide-and-conquer strategy may be required to cope with some particular sentences with unique properties.[5] Lexicon-based techniques and ML-based methods are the two main strategies that are frequently utilized for SA tasks on product evaluations.[6] Sentiment terms are words that express the sentiment of the text document; they are typically verbs and adjectives. Simply, the lexicon-based approach uses a sentiment lexicon to identify the sentiment value of each sentiment term that is extracted from a given text. Many techniques to provide users with a useful methodology for categorizing sentiments have been put forth throughout the years. These strategies have progressed from dictionary-based strategies to systems utilizing ML.[7]

Although SA is viewed as a standard text classification task by ML-based methods, these approaches rely on ML techniques. Utilizing ML techniques, a text classification task divides a piece of text input into a number of predetermined classifications.[8] ML methods are employed to categorize text documents into one of three categories for the SA task: positive, neutral, or negative. ML methods create a model that utilizes the properties of a labeled text that are extracted for a specified set of training text data. After that, unlabeled text is classified utilizing the model. Thus, the robustness of the ML techniques as well as the extracted text characteristics affects the outcome of the supervised SA task.

The majority of recent studies[9,10] on supervised SA focused more on the ML methods already in use than on creating reliable text characteristics. A synopsis of those works can be found in the "Related work" section. Thus, the most difficult tasks in the field of supervised SA is still text feature extraction. This study aims to address the prior identified research gap by improving the outcomes of supervised self-administered photosynthesis SA by developing a robust text feature extraction method. Additionally, the research will evaluate the efficacy of the suggested text features through the use of multiple ML methods and feature selection techniques. Supervised learning is a ML task of learning a function (classifier) using pre-labelled samples as a training dataset. A key step in supervised learning is feature extraction. Traditional ML methods represent text with hand-crafted methods, e.g., n-grams. Recently, DL have been used for automatic feature extraction, including convolutional neural networks (CNNs), recurrent neural networks (RNNs) etc. Describe the fundamentals of CNN, in particular the form that is known as a simplified CNN that has just one convolution layer. The EPLF-TSVM structure is then thoroughly explained for even more precise SA. The research's significance is embodied in the following essential ways:

- Data methods for pre-processing are utilized to eliminate any noise from the provided dataset to improve the text-based data quality. Utilizing word embeddings techniques such as word2vec, the provided dataset is converted from text to numerical data.
- DL techniques For calculating the NSS and PSS, several parameters are computed with CNN and EPLF-TSVM. To categorize the phrases in the data set being utilized into the two labels "negative" and "positive," the classifier EPLF-TSVM is utilized with the results of the preceding stage (NSS and PSS). Increased accuracy has been achieved since the proposed EPLF-TSVM-DL automatically extracts more accurate features and different entities from the provided dataset.

These parts contain the remaining portion of the manuscript. The "Related work" section examines recent research that is relevant to this subject. The suggested approach is explained in the "Proposed method" section. The outcome and the discussion in the "Experimental results and discussion" section are examined in this paper. Lastly, provide a summary of this study's findings in the "Conclusion" section.

Related work

Zarisfi Kermani et al.(11) presented three stages to an ML solution to the TSA issue. The vector space model utilizes a weighted mixture of the values derived from four approaches to get an appropriate value for each feature. A genetic algorithm is employed to solve the optimization problem of determining the percentage of contributions or weights of each approach. As a crucial T-conorm technique, the weighted values from the four approaches are integrated utilizing the Einstein sum. Four well-known Twitter datasets, the Stanford testing dataset, the STS-Gold dataset, the Obama-McCain Debate dataset, and the Strict Obama-McCain Debate dataset are utilized to evaluate the efficacy of the suggested approach according to the accuracy of SVM and multinomial naïve Bayes classification methods. Several issues need to be taken into account, including severe text sparseness and coarse granularity in sentiment analysis.

Zhang et al.(12) suggested a new strategy integrating a popular sentiment categorization technique, SVM with latent SA was suggested, combining cognitive assessment theory with SA. The moderating impact of emotion on the assistance of online reviews under the two categories of items, including the influence of these four types of emotions, happiness, hope, disgust, and anxiety are utilized in this study to categorize the emotions seen in online reviews. Their innate qualities, such their boisterous and informal verbal style, continue to be difficult for many NLP tasks, especially SA.

Song et al.(13) combined the TF-IDF algorithm with SVM provides a chi-square statistic which incorporates the word frequency factor, inter-class concentration coefficient, and correction coefficient to create a Japanese text emotion classification system. In an effort to address the issue of the conventional feature weighting technique TFIDF ignoring the feature item distribution both within and between classes when determining the weight of feature objects, chi-square statistics and intra-class information entropy were implemented to mitigate the impact that TFIDF ignores the distribution of feature items within classes and between classes, respectively. The research creates a controlled test to evaluate the system's functionality. Prior to identifying patterns in text, a more basic step needs to be defined: how automatic techniques can quantitatively represent textual content.

Zainuddin et al.(14) proposed hybrid sentiment categorization for Twitter by the incorporation of a feature selection technique. An analysis is provided comparing the categorization efficiency of the PCA, LSA, and RP feature selection techniques. Utilizing Twitter datasets for illustrating several domains, the hybrid sentiment classification was confirmed, and assessment with various classification methods further proved that the novel hybrid technique yielded significant outcomes. The results of the deployments demonstrated that the novel hybrid sentiment classification managed to outperform the baseline techniques' accuracy by 76,55, 71,62, and 74,24 %, accordingly. Such models' primary drawback is that they frequently have trouble handling OOV words. More sophisticated methods, such FastText, were suggested to remedy this shortcoming.

Xu et al.(15) studied the COVID-19 vaccination discussion on Twitter. In particular, the Twitter API is utilized to gather all COVID-19 vaccine-related tweets from December 15, 2020, to December 31, 2021. Unsupervised learning is then applied. The dataset's sentiment value is determined by applying the VADER system to evaluate the various emotion classes. Following the computation of the number of topics, subjects and keywords are extracted utilizing the LDA framework. People's opinions about the Chinese vaccine differed from those of people in other nations, and the sentiment value may have been influenced by the quantity of daily news deaths and cases, the nature of significant issues in the communication network, the degree and development of ten major public health concerns, and other factors. It also offers perspectives on vaccine trust. People were talking about the Chinese vaccine on Twitter frequently as time went on and vaccinations became more common. However, there hasn't been a noticeable rise in conversations regarding vaccinations in general.

Hidayatet al.(16) executed to assess public opinion regarding this development, which was split into three groups: neutral, pro, and contra. This study utilized two Doc2Vec approaches: the distributed approach and the distributed bag of words, with logistic regression and SVM serving as the classifiers. With an accuracy rate above 75 %, every combination of the simulations and classifier demonstrates that nearly all are opposed to Rinca Island's development. But regardless of the context in which a word will appear, they precompute its representation for every word. These frameworks' static quality causes two issues: (i) they fail to take into consideration the variety of meanings that words can have, and (ii) they have trouble picking up on long-term meaning relationships.

Krishna et al.(17) suggested the SVM Method with Independent Component. To achieve sentiment categorization, the SVM Method utilizes five layers. Utilizing words taken from user review comments, the SVM Technique preprocesses, extracts features, and classifies to improve classification accuracy. To determine the sentiment class label, the semantic opinion words are examined in that layer utilizing the Support Vector Regressive Sentiment Classification in SVM Technique. There are three categories for the sentiment class: good, neutral, and negative. The output layer receives its findings in the end. Nevertheless, this method did not reduce the level of computation.

Solairaj et al.[18] suggested EESNN-SA-OPR technique. P-P similarity and FC are employed as novel recommendation algorithms. The product recommendations are categorized as excellent, good, very good, bad, and very bad employing the augmented Elman spike neural network. The suggested approach is implemented through MATLAB, and various performance metrics, including MAE, MSE, MAPE, accuracy, F-Score, recall, and precision, are employed to evaluate the technique's effectiveness. The EESNN-SA-OPR approach yields a mean absolute error reduction of 12,33 %, 21,31 %, and 41,09 %, and an accuracy increase of 23,14 %, 15,96 %, and 31,54 %. Their natural linguistic traits, such as their informal and loud style continue to be difficult for many NLP tasks, SA considered.

Balaganesh et al.[19] developed SWOANN was utilized by an aspect-based sentiment categorization framework to categorize the sentiment of important features of goods and services. The recommended study improves the classifier's overall efficacy by utilizing important features like the TF-IDF, positive opinion score, and negative opinion score to describe each component of services and products. The appropriate selection of weights of the neurons in the suggested framework has enhanced the processing speed and accuracy of sentiment categorization of the goods and services.

Es-Sabery et al.[20] introduced a novel MapReduce enhanced weighted ID3 decision tree categorization method for OM, primarily comprising three elements: Initially a number of feature extractors, such as N-grams, Bag-Of-Words, word embedding, FastText, and TF-IDF, were utilized to effectively identify and extract the pertinent data from the provided tweets. Secondly, utilized a multiple feature selection such as Gini Index, Gain Ratio, Information Gain, and Chi-square to lower the dimensionality of the high feature. utilized the acquired characteristics in the end to complete the classification task with an enhanced ID3 decision tree classifier that seeks to compute the weighted information gain rather than the information gain observed in conventional ID3.

Inference: depending once more on the format of those outputs, machine learning techniques seem to generally provide the same outcomes. This section maintains the assumption that SA will find additional uses in the future and that systems and services will standardize the utilization of SA methodologies. The suggested additional research will investigate alternative datasets integrating DL and SVM techniques with a focus on three distinct properties. With this research, it can identify concerns related to reputation, collusion, and control, as well as unfair good and negative reviews. In the current research, an innovative EPLF-TSVM-deep learning method is developed, which essentially blends the CNN and TSVM deep learning networks. This method is inspired by the benefits of DL methods in the SA field. as CNN does not manually incorporate the feature extractor during the training process. Specialized neural network types comprise CNN's feature extractor, which determines the weights during training.

METHODOLOGY

To improve the classification rate at the sentence level, an improved EPLF-TSVM-DL is suggested to identify the positive or negative polarity of sentiment sentences. The approach is divided into five basic sections: The first portion consists of pre-processing the data to decrease noise and improve quality; the second part uses word embedding techniques (word2vec) to transform textual input into numerical data. The suggested hybrid approach, which incorporates CNN and EPLF-TSVM in the third part, builds an efficient automated process for extracting features from gathered unstructured data and computing both PSS and NSS. The MFS, an EPLF-TSVM classifier, is employed in the fourth part to categorize the results of the two learning approaches into three classes: Negative and Positive. An experimental evaluation among the recommended EPLF-TSVM-DL and a few additional approaches from the existing research is also conducted to demonstrate the approach's efficacy. The practical outcome demonstrated that the EPLF-TSVM-DL outperforms the other algorithms pertaining of true negative, true positive, false positive, error, false negative, precision, classification, F1-score, kappa statistic, time consumption, convergence, complexity, and stability.

Input data collection details

Two datasets were selected for this research to demonstrate the effectiveness of the developed EPLF-TSVM-DL technique. Sentiment140 is the name of the first dataset. which Twitter's application programming interfaces (API) are utilized to extract. There are 1 600 000 tweets in there that have had their emoticons deleted. The tweets were classified as either positive or negative, with a score of 0 for negative and 1 for positive. It has the following six characteristics: Target, Ids, Date, Flag, User and Text.

The Twitter API is employed to obtain the second dataset, COVID-19_Sentiments. There are 637 978 tweets in total. The tweets were classified as belonging to two classes: positive-1 and negative 0.[21] It has the following qualities: Target, IDs, Time, Place, and Text: the user-posted text. The emotive score and text qualities are critical to our job. All other features were eliminated for that. Moreover, the dataset's target distribution exhibits an imbalance, with 120 646 negative tweets, 259 458 neutral tweets, and 257 874 positive tweets. A total of 574 182 tweets were utilized in the learning process for training, while 63 796 tweets were utilized for testing. Here, 30 % is utilized for testing and 70 % are employed for training.

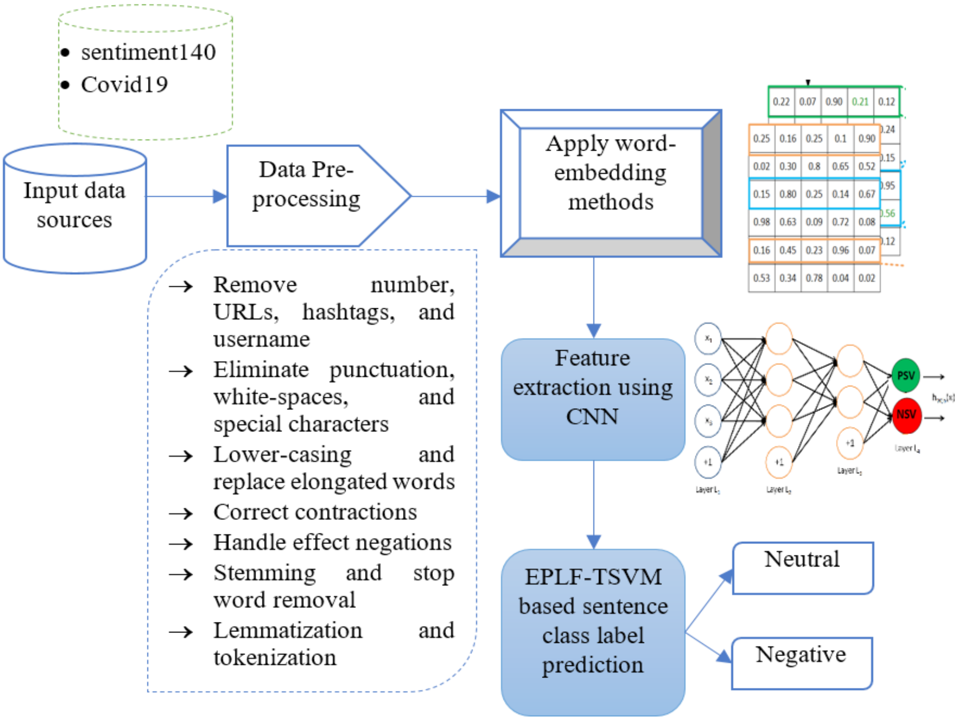
sentiment140
Covid19
Input data sources
Data Pre-processing
Apply word-embedding methods
→ Remove number, URLs, hashtags, and username
→ Eliminate punctuation, white-spaces, and special characters
→ Lower-casing and replace elongated words
→ Correct contractions
→ Handle effect negations
→ Stemming and stop word removal
→ Lemmatization and tokenization
Feature extraction using CNN
PSV
NSV
EPLF-TSVM based sentence class label prediction
Neutral
Negative

Figure 1. Architecture of the proposed EPLF-TSVM-DL approach for Sentence-Level Classification

Data Pre-Processing Stage

Selecting appropriate and efficient pre-processing methods might improve classification accuracy because pre-processing tasks are considered the initial stage of the text classification problem. Preparing, normalizing, removing, and cleaning the noisy data from the given dataset that will be classed is the main objective of the text pre-processing method. The subsequent pre-processing techniques employed in this study are explained as follows:

Remove number, URLs, hashtags, and username: since numbers, URLs, hashtags, and usernames don't convey any emotion, it's standard procedure to remove them from the pre-processing statement.

Eliminate punctuation, white-spaces, and special characters: The stop, question, and exclamation marks are the three punctuations that need to be eliminated from the tweet after eliminating all white space. Since none of the discovered unique characteristics have an effect on the sentiment represented, they are all eliminated. Just the capital and lowercase characters were retained at this point following all of the pre-processing methods mentioned earlier.

Lower-casing: All characters that aren't letters were removed using the earlier outlined procedures. Thus, the lower casing comes next. To put it another way, every letter that was retained in the tweet was changed to lower case, which decreased the number of dimensions in each word. Take the place of long words: With this process, the letter that appears in the elongated word, such as "haaaaaaappy," at least three times, is removed. Following this technique, the word is normalized to at most two characters and becomes "happy."

Correct contractions: The pre-processing method can include the correction of contractions as one strategy. For instance, the corrected version of the words "aren't" and "weren't" would be "is not" and "were not," respectively.

Handle effect negations: This method substitutes the term that comes before NOT with its antonym. The antonym denotes the substitute word's opposing connotation. This method looks for words that are preceded by NOT in each tweet. If a word is found to have an antonym in the WordNet dictionary, it substitutes the original term with its clear synonym. For instance, it substitutes the word "beautify" for the phrase "not uglify."

Stemming and Remove stop-words: is the process of condensing multiple words into one to make words smaller. This method removes a word's endings in order to find the word stem in a dictionary. The terms that appear frequently in the tweeted message are known as stop-words. They are eliminated because it is thought unnecessary to deal with them because they are emotionless. As a result, utilizing the established stop-words list, all stops-words discovered in the tweet are eliminated in this research.

Tokenization: The technique of tokenization divides phrases into units called tokens. Longer evaluation information paragraphs can be divided into sentences throughout this method. After that, these sentences might be divided into tokens.

The study applies all previously stated strategies to the dataset provided. Additionally, it creates a lookup database with 3 000 words and phrases that comprise words, slang terms, and abbreviations so that the incorrect words can be substituted for the slang and abbreviations in the tweet that is presently being analyzed. Users of the Twitter network frequently make typographical and spelling errors, which could complicate the learning process. Thus, employ Norvig's spelling and typographical corrector, which corrects them automatically, to increase the efficiency of the learning process. Following the pre-processing stage, which eliminates noisy data from the dataset, word embeddings are the next step, as explained in the subsection that follows. As a result, word embedding methods will utilize data from the program of all pre-processing methods as input.

Word embeddings

Only numerical input can be processed by the CNN deep learning method. Text-based data that was acquired during the pre-processing data phase utilizing deep-learning algorithms needs to be converted into numerical-based data in addition to be utilized in proposals. The key problems in NLP is this operation, which is known as vectorization. For the larger datasets, GloVe, Word2vec, and Fast text introduced by Stanford, Google, and Facebook, prove to be effective approaches.(22) Thus, word embeddings data utilizing Word2vec approaches come next following the pre-processing data phase.

Word2Vec: At this point, the preprocessed data is utilized in the word embedding procedure. The Gensim package is employed for executing Word2vec for the word embedding procedure. This procedure generates an instance as its output after receiving the preprocessed data in a format of tokenized texts. This representation has the shape of a word vocabulary with vectors connected to every word. The variables chosen during simulation determine the number of vector dimensions. The Word2vec model is trained utilizing the following variables: size (the number of vector dimensions is 100), window (the size of the context word window is 2), workers (there are four concurrent threads), min_count (the minimum word count is 1), and iter (there are forty training iterations). When a target word is entered into Gensim's "most_similar" function, ten words that are the closest to the target word are returned. Researchers selected the word "jelek," which translates to "poor quality," to test the algorithm. The "most_similar" function provides 10 terms that are somewhat similar in significance to the word "jelek," which is an optimistic outcome for the framework.

Feature Extraction Using CNN

As demonstrated by the pseudocode in table 2, the primary goal of CNNs is to develop a method for reducing the overall number of variables and building a deeper neural network with fewer variables. The general architecture of CNNs is shown in figure 2, where the three basic layers are the pooling layer, convolution layer, and fully connected layer.

Convolution Layer is the basis of CNN and is always the top layer in the network's general architecture. The main goal of this layer is to employ one of the word embedding techniques on the provided input sentence to identify and gather features from a produced matrix. A convolved feature is produced by the convolution layer by sliding a filter over the embedding matrix. To generate numerous feature maps, the embedding matrix is subjected to several filters. In the intermediate job that connected the convolution layer and the pooling layer, these acquired feature maps are activated (the linear feature maps are transformed into non-linear feature maps) utilizing the Rectified Linear Unit (ReLU) activation function. Ultimately, the pooling layer receives the acquired non-linear feature maps. In conclusion, the greatest utilized activation function with the convolution layer is the sigmoid function. All it does is compute with this formula (1):

$$f(y) = \max(o, y) = \frac{1}{1+e^{-g}}$$

Where g denotes the advanced word embedded vector created by the word2vec. As seen in figure 3, the activation function really outputs 0 if it receives a negative value as input and the same positive value *af*(24) if it receives a positive value.

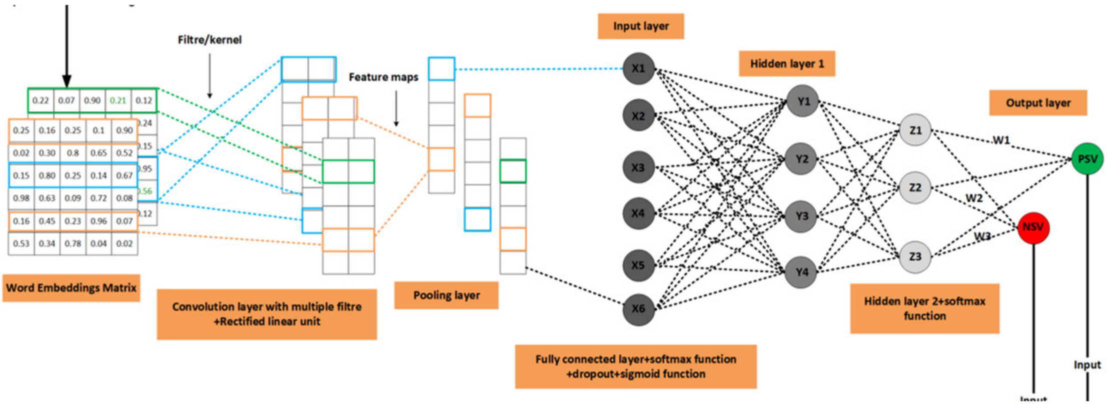

Figure 2. Overall structure of the CNN based feature extraction

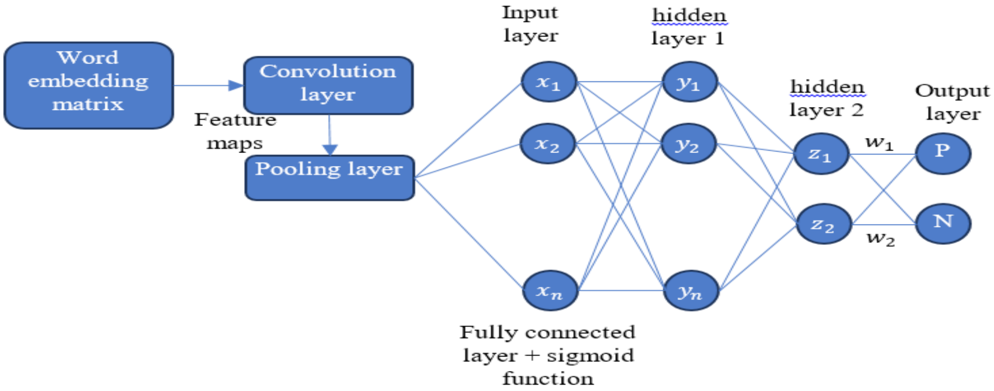

Figure 3.

Compared to other activation techniques like tanh or sigmoid, the sigmoid function has the advantage of being able to avoid the vanishing gradient issue, having a comparatively low execution time, and having faster convergence due to its basic math equation.

Pooling Layer: Following the first phase of convolving the embedding matrix with numerous filters, the pooling layer is applied in the second phase to lower the dimensionality of the feature maps that were created in the initial step. Consequently, the overfitting issue is limited, the total number of CNN variables is reduced, and the expense of computing is reduced. The average and maximum pooling operations are two common pooling functions. The pooling feature in the convolved feature map is determined by the average of all the values, according to the average-pooling approach. The convolved feature map's maximum component is chosen as the pooling feature via the max-pooling procedure, which discards the remaining elements. Utilized the max-pooling technique in this research. Convolutional feature maps are typically transformed into a single column by the pooling layer before being handed on to a fully connected layer.[25]

Fully Connected Layer is another name for a dense layer, which is employed to determine the sentimental scores of each input sentence in terms of PSS and NSS utilizing the single column that was received from the pooling layer in the preceding stage. The operation summed up as a linear process where each input is connected to every output by a distinct weight.[25] Equation (2) is employed by the fully connected layer to compute the sentimental values (SV) in the following manner:

$$SV = af(WM * SCPL + bias)$$

Where is sigmoid activation technique, *WM* is utilized weight matrix, *SCPL* is the single column attained from the pooling layer, and *bias* is the bias. In the intermediate learning phase among the fully connected and output layers, the sigmoid activation function known as a normalized exponential function is employed. The completely connected layer's numerical values are converted by this activation function into probable values that fall between 0 and 1, with the total of these probable values equal to 1. In this case, the obtained vector of z real values through the EPLF-TSVM was subjected to the softmax function utilized to determine two values: positive and negative sentimental scores. Equation (3) shows the notation for a softmax activation function:

$$f\ (y) = \frac{\exp_i^y\, af}{\sum_{k=1}^{K} exp_i^y\, af}$$

Where *af* is softmax activation technique, y is the input value, *exp_i^y* is input value's standard exponential function, and K is the quantity of classes present in the dataset.

Table 2. Pseudocode of feature extraction using CNN

```
Input: Datasets D, word2vec
Output: Feature sets
Let feature be the feature set matrix
For i in D do
Convolution layer←obtained non-linear feature maps and do
f(y)=max(o,y)
Apply the max-pooling←convolved feature maps to a single column
Calculates the sentimental values (SV)
Finalize vector v_j←vectorize(j,word2vec)
Add on v_j to feature
feature_train,feature_test←split feature set
Return feature
```

Sentence level classification using EPLF-TSVM

Strong supervised ML methods called support vector machines (SVMs) are frequently employed to address problems with regression. SVM maximizes the margin among the data samples of two classes by identifying the ideal separation hyperplane. SVM are much harder to understand in higher dimensions. To get around this problem, Jayadeva et al.[27] developed TSVM, which locates two nonparallel hyperplanes with the intention of placing each one as close as possible to samples from one class and as far away as feasible from data from the other class. Determining the decision border and the way the data might be divided linearly are much more challenging to visualize. Before creating a support vector classifier, data must first be transformed into a higher-dimensional space because, they are rarely linearly separable. Nevertheless, the pinball loss function SVM is quite sluggish and has a significant computational complexity for large-scale applications. This section introduces the EPLF-TSVM, a unique classification method that combines the TSVM framework with a truncated pinball loss function. The objective of the next section is to address the core of the suggested approach, which is to minimize the computational difficulty by presenting both the linear and nonlinear cases of a TSVM with an extensive pinball loss function that is targeted into the binary classification issue. These cases yield the best outcomes when error minimization and the classifier's capacity to generalize are simultaneously maximized.

The symmetric kernel method is suitable to fix this issue. The linear EPLF-TSVM is now extended to the nonlinear case utilizing a symmetric kernel technique.[26] The effectiveness of EPLF-TSVM is largely dependent on the symmetric kernels that are employed. If *KF(.)* is the specified kernel function, then the nonparallel hyperplanes in the kernel-generated space are as follows.

Where $weight^1$, $weight^1 \in R^m$, and:

$$X = \begin{bmatrix} A_{m_1 \times n} \\ B_{m_{1\times n}} \end{bmatrix}$$

The EPLF is a loss function that is frequently employed to train classifiers in the nonlinear case of issues with adding slack vector ϑ and c_1,c_2 positive penalty parameters and v_1 and v_2 are vectors of appropriately sized ones.

$$\min_{weight^1,bias^1,\vartheta} \frac{1}{2} \|KF(A,X^T)weight^1 + +bias^1\| + c_1\, v_2^T\, \vartheta \quad (2)$$

$$\min_{weight^2,bias^2,\vartheta} \frac{1}{2} \|KF(A,X^T)weight^2 + +bias^2\| + c_2\, \mathrm{v}_1^T\, \vartheta \quad (3)$$

The Lagrange function is implemented, and the Karush–Kuhn–Tucker (KKT) optimality requirements are utilized to produce the dual of (4). For this, introduce the n1,n2,τ1,τ2 non-negative parameters and Lagrange multipliers δ≥0, ε≥0 , σ≥0 and φ≥0.

$$\min_{\delta,\varepsilon} \frac{1}{2}(\delta-\varepsilon)^T \mathfrak{Q}(\mathfrak{F}^T\mathfrak{F})\mathfrak{Q}^T(\delta-\varepsilon) - (\delta-\varepsilon)^T v_2 \left(1+\frac{\tau_1}{\mathfrak{y}_1}\right) + \delta^T v_2 \left(\frac{\tau_1}{\mathfrak{y}_1}+\frac{\tau_2}{\mathfrak{y}_2}\right) \quad (4)$$

Eq. (4)'s dual function may be found in this manner:

$$\min_{\sigma,\varphi} \frac{1}{2}(\sigma-\varphi)^T \mathfrak{F}(\mathfrak{Q}^T\mathfrak{Q})\mathfrak{F}^T(\sigma-\varphi) - (\sigma-\varphi)^T v_2 \left(1+\frac{\tau_1}{\mathfrak{y}_1}\right) + \sigma^T v_2 \left(\frac{\tau_3}{\mathfrak{y}_3}+\frac{\tau_4}{\mathfrak{y}_4}\right) \quad (5)$$

Where:

$$\mathfrak{F} = [KF(A,X^T) \quad v_1],\ \mathfrak{Q} = [KF(B,X^T) \quad v_2].$$

Ultimately, the finest separating hyperplanes are given by:

$$\begin{bmatrix} weight^1 \\ bias^1 \end{bmatrix} = -(\mathfrak{F}^T\mathfrak{F} + \gamma I)^{-1}\mathfrak{Q}^T(\delta-\varepsilon) \qquad (6)$$

$$\begin{bmatrix} weight^1 \\ bias^1 \end{bmatrix} = -(\mathfrak{F}^T\mathfrak{F} + \gamma I)^{-1}\mathfrak{Q}^T(\sigma-\varphi) \qquad (7)$$

Since we cannot ensure F^TF and Q^TQ are irreversible, it is usually positive semi-definite, although it might not be well conditioned in some situations. To take the potential for improper conditioning of F^TF and Q^TQ, the regularization term **γI**(γ>0) must be utilized. A new sample point x∈Rn is allocated to class i(i=+1 or -1) by the new sample point $x \in R^n$ depending on which of the two hyperplanes it lies closest toward.

$$y = class(i) = \arg\min_{i=1,2} \frac{|x^T weight^i + bias^i|}{\|weight^i\|} \qquad (8)$$

Experimental Results and discussion

This section describes the experimental outcomes of the EPLF-TSVM-DL. These findings are produced by employing the EPLF-TSVM-DL classifier and other literature methods such as SVM,(13) EESNN-SA-OPR and SWOANN(18,19) on both utilized datasets, as shown in the subsection on data collection. The provided dataset was often divided into two categories for this research: a training dataset, which comprised 70 % of the total dataset, and a testing dataset, which comprised 30 % of the entire dataset. Next, save the training and testing datasets that were gathered. After storage is complete, minimize and eliminate noisy data from training and testing datasets utilizing a variety of text pre-processing methods. Subsequently, employ an incredibly effective word embedding technique to convert the textual data into numerical data. Additionally, implement EPLF-TSVM-DL classifier on the testing dataset. The suggested approach employs accuracy, recall, precision, and F1-score as metrics to assess the results of the categorization of SA and ED data. Accordingly, the assessment metric utilized to gauge the overall outcome is called accuracy. Four separate categories might be utilized for categorizing the evaluation criteria that were previously explained: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN). The acronym TP stands for the number of positive-emotion samples that were correctly classified as positive samples. The word "FP," indicates the quantity of samples that were incorrectly anticipated to be positive but really contained negative feelings. The percentage of negative emotion samples which were accurately projected to be negative is denoted by TP, FP, and TN, while the number of positive emotion samples which were mistakenly forecasted to be negative is represented by FN. Additionally, accuracy is determined by the evaluation criterion Equation (9)'s performance matrix as the percentage of properly predicted results, which is expressed as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} * 100$$

$$Precision = \frac{TP}{TP + FP} * 100$$

$$F1\ score = \frac{2 * Precision * Recall}{Precision + Recall}$$

$$Specificity = \frac{TN}{TN + FP} * 100$$

$$Sensitivity = \frac{TP}{TP + FN} * 100$$

The efficacy of Word2vec was assessed in this study with respect to emotion recognition (ER) and TC. To validate and confirm the results, these research' word embedding techniques were combined with the recommended EPLF-TSVM-DL. The practice of classifying texts into predefined groupings is known as text classification (TC). The numerical results of proposed and exiting method for both sentiment-140 and COVID-19-sentiments are illustrated in table 1 and 2.

Table 1. The numerical results of proposed and exiting method for both sentiment-140

Metrics	Svm	Eesnn-Sa-Opr	Swoann	Eplf-Tsvm-Dl
Accuracy	84,1400	88,5600	94,0500	97,0788
Precision	85,2500	87,7700	94,9000	97,0816
Recall	82,3400	85,6700	93,9000	96,7751
F-measure	83,1900	86,7700	94,8900	96,9281

Table 2. The numerical results of proposed and exiting method for both COVID-19-sentiments

Metrics	Svm	Eesnn-sa-opr	Swoann	Eplf-tsvm-dl
Accuracy	80,1400	83,5600	89,1500	93,1571
Precision	81,2200	83,4700	85,7000	93,8502
Recall	80,3400	83,6700	84,9000	92,2149
F-measure	80,2900	83,2700	84,8900	93,0253

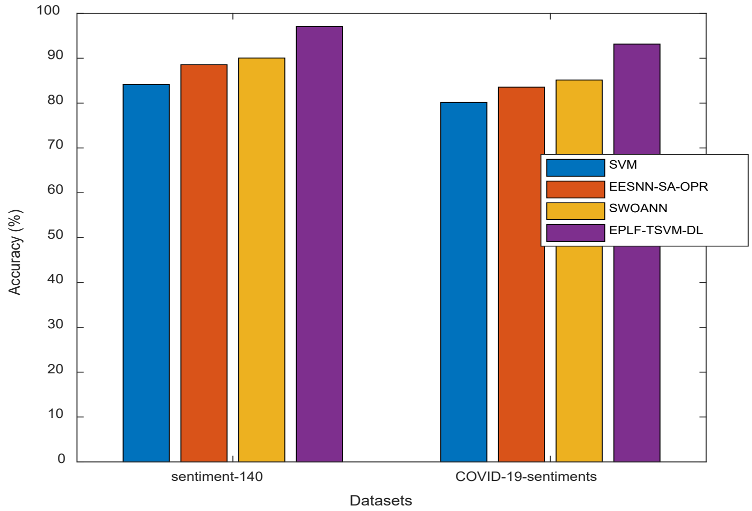

Figure 4. Accuracy performance comparison

The accuracy of suggested and present framework for the quantity of features of a given database shown in figure 4. With a score of 93,1571 % for COVID-19 sentiments and 97,0788 % for sentiment140, the suggested EPLF-TSVM-DL improves accuracy. Because the threshold is mostly utilized to change the size of sub-training

https://doi.org/10.56294/sctconf2024702

datasets, this is the case. The probability that the examples in the raw dataset will be distributed across the sub-training datasets increases with decreasing value. However, for the cluster number, trial and error are required to determine the optimal value across the various datasets. Because it remembers long-range links and dependencies inside the phrases, including an attentional process improves accuracy efficiency in the suggested framework.

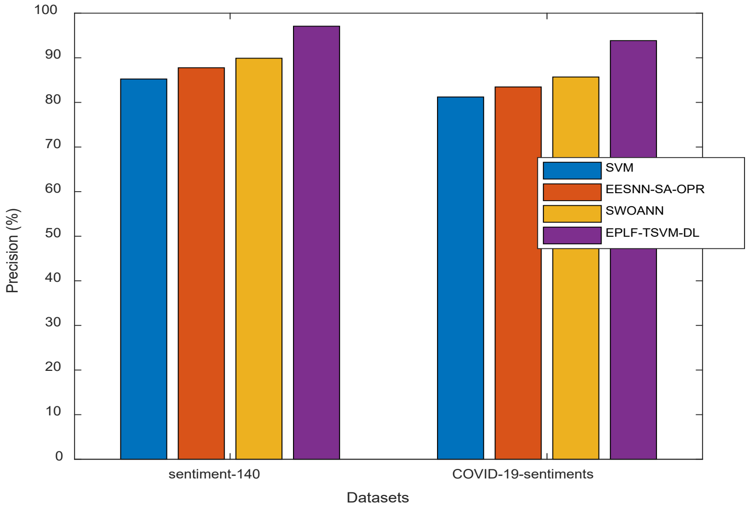

Figure 5. Precision performance comparison

The precision of suggested and present framework for the quantity of features in a particular database, such as sentiment 140 and COVID-19 sentiments, may be seen in figure 5. A higher feature measure results in more precision. When compared to the SVM, EESNN-SA-OPR, and SWOANN, the proposed EPLF-TSVM-DL achieves a precision of 97,0816 % and 93,8502 % for sentiment 140 and COVID-19 sentiments, respectively. This is because EPLF-TSVM-DL reduces the time it takes to compute the derived factors, making fine-tuning EPLF-TSVM-DL easier and increasing the precision rate. The connection between various sections of the phrases is better preserved in the framework of subjectivity analysis, which helps to connect things and opinions that are far apart and so improves outcomes.

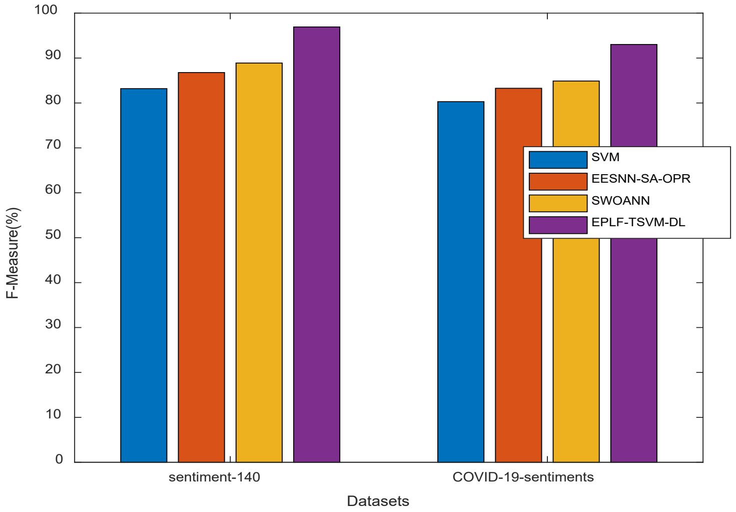

Figure 6. F-Measure performance comparison

Figure 6 illustrates the F-measure of the existing and suggested frameworks for the quantity of features in the designated databases, such as sentiment 140 and COVID-19 sentiments. Along with the quantity of characteristics, the f-measure is also enhanced. When compared to other frameworks such as SVM, EESNN-SA-OPR, SWOANN and EPLF-TSVM-DL delivers an f-measure of 96,9281 % and 93,0253 % for sentiment 140 and COVID-19 sentiments, respectively. The reason for this is that the clustering will reduce the computational complexity of the ACNN-BILSTM learning in an effective manner resulting in strong evaluation results and a high

F1-score rate for the ACNN-BILSTM. In sentence-based subjectivity analysis, the combined power of CNN to enhance the classification rate and prevent the persistent issue.

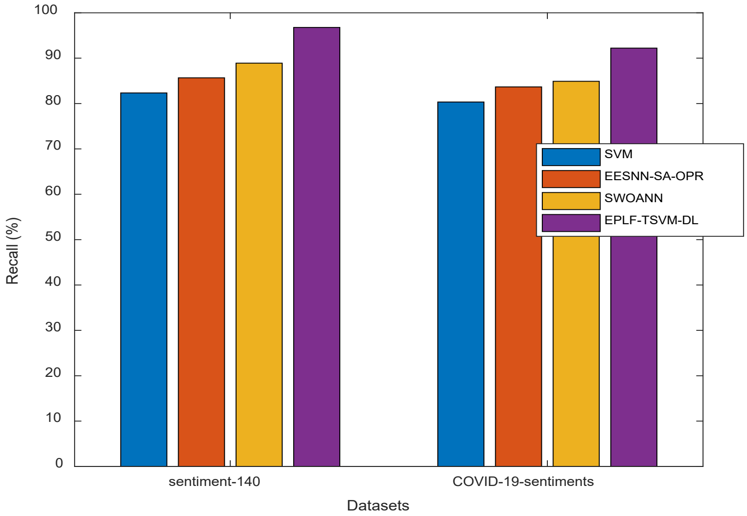

Figure 8. Recall performance comparison

The recall of suggested and present framework for the quantity of characteristics in a database, such as sentiment 140 and COVID-19 sentiments, is shown in figure 8. A higher feature measure results in a higher recall. When compared to the SVM, EESNN-SA-OPR, SWOANN and EPLF-TSVM-DL achieves a recall of 96,7751 % for sentiment 140 and 92,2149 % for COVID-19 sentiments, respectively. Existing techniques are underfitting because they are simplistic models that are inadequate for high-dimensional datasets. The suggested EPLF-TSVM-DL rate is improved by using CNN feature extraction, which avoid computational complexity. Because the CEL function is beneficial to the model, by lessening the effect of class imbalance on the training procedure and encouraging the algorithm to concentrate more on classes that are more difficult to identify, it can enhance efficiency by enabling the framework to provide its best output.

CONCLUSION

The variety of social media sites encourages individuals to engage with them extensively, viewing them as effective means of communication. Consequently, the large SA data to be learned has been produced by user feedback on these sites. With more data being produced, the network environment is getting more complicated. Due to its importance in comprehending public opinion, natural language processing researchers have made emotion detection and SA research a top focus. Because artificial intelligence is still developing, precise emotion recognition and SA now have a great scientific accuracy. The study focuses on Urdu because it is a low-resource language that has not received much attention in the past ten years when it comes to emotion detection and SA in Twitter. Furthermore, most low-level language issues, like the ones for which deep learning techniques are employed in ED and SA, have low analytical accuracy because there isn't a publicly available corpus. The suggested approach integrates CNN with EPLF-TSVM to create the EPLF-TSVM-DL framework. The reason for this is that while CNN retrieves deep features, the CNN hidden layer is dependent on the results of the prior session. Sentiment140 and COVID-19_Sentiments datasets were selected during the data gathering phase to assess the proposal. The first experiment was conducted to assess the efficacy of the performed data pre-processing tasks on the two datasets utilized in the data pre-processing stage. Multiple pre-processing tasks were done to both datasets. Subsequently, the Word2Vec method was utilized for data representation, converting textual data into numerical data that included word embedding. Utilizing the corpus gathered for SA and emotion recognition, the accuracy is increased. Therefore, by employing the corpus gathered for sentiment and emotion recognition, the accuracy is improved. In the future, suggest employing a bidirectional LSTM network in place of the CNN to increase the analytical effectiveness of the framework. Future work will combine the concept with fuzzy logic theory to handle continuous-valued features while accounting for various variables related to feature pickers and extractors. SA utilizing the Mamdani fuzzy system as a classifier to address ambiguity and uncertainty in the sentiments conveyed in the data shared by social media users. Fuzzy rule-based framework integration with the enhanced ID3 decision tree from MapReduce for interpreting voice signals on social media platforms.

REFERENCES

1. Liu B. Sentiment analysis: Mining opinions, sentiments, and emotions. Cambridge university press. Pp. 1-78.

2. Kumar A, and Sebastian TM. Sentiment analysis: A perspective on its past, present and future. International Journal of Intelligent Systems and Applications, 4(10), pp.1-14. DOI: 10.5815/ijisa.2012.10.01.

3. Hussein DMEDM. A survey on sentiment analysis challenges. Journal of King Saud University-Engineering Sciences, 30(4), pp.330-338. https://doi.org/10.1016/j.jksues.2016.04.002.

4. Cambria E, Schuller B, Xia Y, and Havasi C, et al. New avenues in opinion mining and sentiment analysis. IEEE Intelligent systems, 28(2), pp.15-21. DOI: 10.1109/MIS.2013.30.

5. Gidiotis A, and Tsoumakas G. A divide-and-conquer approach to the summarization of long documents. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 28, pp.3029-3040. https://doi.org/10.48550/arXiv.2004.06190.

6. Erşahin B, Aktaş Ö, Kilinç D, and Erşahin M, et al. A hybrid sentiment analysis method for Turkish. Turkish Journal of Electrical Engineering and Computer Sciences, 27(3), pp.1780-1793. DOI 10.3906/elk-1808-189.

7. Gadri S, Chabira, S, Ould Mehieddine S, and Herizi K, et al. Sentiment analysis: developing an efficient model based on machine learning and deep learning approaches. In Intelligent Computing & Optimization: Proceedings of the 4th International Conference on Intelligent Computing and Optimization 2021 (ICO2021); 3, pp. 237-247.

8. Bakalos N, Papadakis N, and Litke A, et al. Public perception of autonomous mobility using ML-based sentiment analysis over social media data. Logistics, 4(2), pp. 1-14. https://doi.org/10.3390/logistics4020012.

9. Wadawadagi RS, and Pagi VB. Sentiment analysis on social media: recent trends in machine learning. Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines, pp.780-799.DOI: 10.4018/978-1-6684-6303-1.

10. Jain PK, Pamula R, and Srivastava G, et al. A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews. Computer science review, 41, pp. 100413. https://doi.org/10.1016/j.cosrev.2021.100413.

11. Zarisfi Kermani F, Sadeghi F, and Eslami E, et al. Solving the twitter sentiment analysis problem based on a machine learning-based approach. Evolutionary Intelligence, 13, pp.381-398. DOI: 10.1007/s12065-019-00301-x.

12. Zhang W, Kong SX, Zhu YC, and Wang XL, et al. Sentiment classification and computing for online reviews by a hybrid SVM and LSA based approach. Cluster Computing, 22, pp.12619-12632. https://doi.org/10.1007/s10586-017-1693-7.

13. Song G. Sentiment analysis of Japanese text and vocabulary learning based on natural language processing and SVM. Journal of Ambient Intelligence and Humanized Computing, pp.1-12. https://doi.org/10.1007/s12652-021-03040-z.

14. Zainuddin N, Selamat A, and Ibrahim R, et al. Hybrid sentiment classification on twitter aspect-based sentiment analysis. Applied Intelligence, 48, pp.1218-1232. https://doi.org/10.1007/s10489-017-1098-6.

15. Xu H, Liu R, Luo Z, and Xu M, et al. COVID-19 vaccine sensing: Sentiment analysis and subject distillation from twitter data. Telematics and Informatics Reports, 8, pp. 100016. https://doi.org/10.1016/j.teler.2022.100016.

16. Hidayat THJ, Ruldeviyani Y, Aditama AR, Madya GR, Nugraha AW, and Adisaputra MW, et al. Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM and logistic regression as classifier. Procedia Computer Science, 197, pp. 660-667. https://doi.org/10.1016/j.procs.2021.12.187.

17. Krishna MM, Duraisamy B, and Vankara J, et al. Independent component support vector regressive deep learning for sentiment classification. Measurement: Sensors, 26, pp. 100678. https://doi.org/10.1016/j.measen.2023.100678.

18. Solairaj A, Sugitha G, and Kavitha G, et al. Enhanced Elman spike neural network based sentiment analysis of online product recommendation. Applied Soft Computing, 132, pp. 109789. https://doi.org/10.1016/j.asoc.2022.109789.

19. Balaganesh N, and Muneeswaran K. A novel aspect-based sentiment classifier using whale optimized adaptive neural network. Neural Computing and Applications, pp. 1-10. https://doi.org/10.1007/s00521-021-06660-w.

20. Es-Sabery F, Es-Sabery K, Qadir J, Sainz-De-Abajo B, Hair A, García-Zapirain B, and De La Torre-Díez I, et al. A MapReduce opinion mining for COVID-19-related tweets classification using enhanced ID3 decision tree classifier. IEEE Access, 9, pp. 58706-58739. DOI: 10.1109/ACCESS.2021.3073215.

21. Rezaeinia SM, Rahmani R, Ghodsi A, and Veisi H, et al. Sentiment analysis based on improved pre-trained word embeddings. Expert Systems with Applications, 117, pp. 139-147. https://doi.org/10.1016/j.eswa.2018.08.044.

22. Goldberg Y, and Levy O. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722, pp. 1-5. https://doi.org/10.48550/arXiv.1402.3722.

23. Ide H, and Kurita T. Improvement of learning for CNN with ReLU activation by sparse regularization. In international joint conference on neural networks (IJCNN), pp. 2684-2691. DOI: 10.1109/IJCNN.2017.7966185.

24. Agarap AF. Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375, pp. 1-7. https://doi.org/10.48550/arXiv.1803.08375.

25. Tanveer M, Sharma A, and Suganthan PN, et al. General twin support vector machine with pinball loss function. Information Sciences, 494, pp. 311-327. https://doi.org/10.1016/j.ins.2019.04.032.

26. Khemchandani R, and Chandra S. Twin support vector machines for pattern classification. IEEE Transactions on pattern analysis and machine intelligence, 29(5), pp. 905-910. DOI: 10.1109/TPAMI.2007.1068.

FINANCING

No financing.

CONFLICT OF INTEREST

None.

AUTHORSHIP CONTRIBUTION

Conceptualization: Savitha D.
Data curation: Sudha L.
Formal analysis: Savitha D.
Research: Sudha L.
Methodology: Sudha L.
Drafting - original draft: Savitha D.
Writing - proofreading and editing: Savitha D.