

¹ Mr. Bikshapathy
Peruka

²Dr. K. Shahu
Chatrapati

Enhancing Sentiment Analysis Through Deep Learning: Leveraging Emotionnet with Improved Mutual Information for Advanced Sentiment Classification



Abstract: - Sentiment analysis is a dynamic implement for learning understandings from textual data across miscellaneous fields, with social media, customer evaluations, and market trends. This research proposal announces EmotionNet, an advanced deep learning background designed to improve the accuracy and efficiency of sentiment analysis. In the preprocessing stage, advanced text cleaning methods are employed, leveraging transformer-based models to standardize and eliminate noise from the text. This involves homogenizing non-standard text, such as contractions and slang, while eliminating irrelevant symbols. Tokenization and lemmatization further refine the text, contravening it into meaningful units and an improved analysis of minimizing words to their base forms. Word embeddings like Word2Vec capture semantic information. Feature abstraction exaggerates text analysis by extracting emotion and contextual embeddings, incorporating syntactic features like dependency parse trees, and proposing novel semantic features based on frequency-inverse document frequency (TF-IDF) and pointwise mutual information (PMI) to capture word co-occurrence patterns. Sentiment lexicons and traditional textual features are also integrated. To augment efficiency, Feature Selection employs advanced techniques, including an improved mutual information (IMI) approach to select the most informative features for sentiment analysis. The classification of sentiment is an innovative architecture of deep learning, combining of convolutional neural network (CNN), and attention-recurrent neural network (A-RNN). An architecture excels were capturing both local and contextual sentiment. Emotion Net represents a significant advancement in sentiment analysis, offering superior accuracy and contextual understanding without compromising efficiency. The proposed model is implemented using Python. The model's performance is notable, boasting an impressive accuracy rate of 95.7% in experiments and tests.

Keywords: Sentiment Analysis; Emotion Net; TF-IDF; PMI; IMI; CNN; A-RNN.

I. INTRODUCTION

The field of processing in natural language and computational linguistics on an empathetic and take out sentiment, emotions, and independent information from text data. In a world submerged with digital content, sentiment analysis has gained prominence valuable tool for individuals, productions, and researchers [1, 2]. It provides insights into public opinion, customer response, and user sentiment, which can inform decision-making, product expansion, marketing approaches, and much more. A universe of the internet and social media stages are finished. It is easier for the public to express their sentiments and feelings online. This surge in user-generated content, including social media posts, product assessments, forum discussions, and news articles, has created a wealth of textual data that contains valuable sentiment-related information [3, 4]. Sentiment analysis techniques leverage this data to automatically classify and interpret sentiments as positive, negative, neutral, or even more fine-grained emotional states [5]. In recent years, deep learning (DL) is a division of machine learning has revolutionized sentiment analysis. Deep neural networks (DNN) as CNNs and recurrent neural networks (RNNs) represent as conventional of an extraordinary proficiency of an understanding of the human language and nuances of sentiment [6, 7]. An advancement of this technology has enabled sentiment analysis to reach new heights of precision and complexity.

In this study of an investigation through DL, methods, applications, and challenges [8] were discovered. An investigation of an evolution DL, and cooperative connection between two. Furthermore, an exploration of real-world applications, indicating the important investigation has been studied [9, 10]. The main goal of sentiment analysis is to robotically discern these sentiments from text data. It involves two main components; text data must be pre-processed to make it amenable to analysis. This includes tokenization, eliminating stop words, stemming or lemmatization, and more. Once the text is pre-processed, sentiment analysis algorithms are classified as positive, negative, or neutral. More advanced analysis can categorize sentiments into a spectrum, allowing for fine-grained analysis [11]. DL has redefined the landscape of sentiment analysis. DNN inspired the human brain

¹*Corresponding author: Research Scholar, Department of Computer Science and Engineering, Jawaharlal Nehru Technological University, Kukatpally, Hyderabad, India. Email: peruka.bpathy@gmail.com

²Professor, Department of Computer Science and Engineering, Jawaharlal Nehru Technological University, Kukatpally, Hyderabad, India. Email: shahujntu@gmail.com

structure, exceptionally compatible for processing sequential data like text. They have the capacity to inevitably absorb complex patterns and illustrations from data, eliminating the need for handcrafted features [12, 13].

Sentiment analysis has greatly benefited from the competencies of DL, and, in turn, it has become a projecting application of DNN. The synergy between the two fields has led to several key developments: DL representations have consistently outperformed traditional approaches, achieving state-of-the-art accuracy in sentiment classification tasks. DL allows sentiment analysis models to discern more coarse sentiments, such as specific emotions (e.g., joy, anger, sadness), making an invaluable for miscellaneous applications [14]. DL models can adapt to different domains, languages, and contexts with minimal manual intervention, making them highly versatile. DL models can process and analyze text data in real time, enabling applications like social media monitoring and chatbot interactions. DL models capture the contextual nuances of language, enabling better recognition of sarcasm, irony, and sentiment shifts within a text [15]. The following terms for the research are used:

- Through innovative techniques using transformer-based models, EmotionNet preprocesses text by normalizing abbreviations, misspellings, and slang, ensuring consistency, and de-noising to remove irrelevant symbols, thereby improving data quality for sentiment analysis.
- Leveraging emotion and contextual embeddings, syntax-based features like dependency parse trees, and novel semantic features based on TF-IDF and PMI enrich feature representation, capturing nuanced sentiments in text.
- Advanced feature selection techniques, notably an IMI approach, are introduced to identify the most informative features for sentiment analysis. This contributes to a streamlined and effective sentiment analysis process.
- A novel deep learning architecture, combining CNN, and RNN with attention mechanisms, captures both local and contextual sentiment, and a tailored loss function for multi-label sentiment classification is proposed, pushing the boundaries of sentiment analysis accuracy and efficiency.

The research article was arranged in the following sequences: Section 2 discusses the literature survey and reviews. Section 3 talks about sentiment classification using EmotionNet: architectural description. Section 4 manifests the recorded results and the final section are conclusion of the research article.

II. RELATED WORKS

In 2021, Kardakis *et al.* [16] proposed that sentiment analysis has gained popularity among researchers due to the advent of social media and source of the internet. It is crucial due to an abundance of opinion-heavy user-generated material, such as product reviews and event opinions. The computer-based commons' views, spirits, and attitudes surrounding products, facilities, and proceedings are greatly wanted after by corporations and provision benefactors.

In 2020, Song *et al.* [17] used probabilistic linguistic term sets (PLTSs) for sentiment analysis of brief texts. Every word in this paradigm is represented by a PLTS that completely describes its polarity of emotion. For both supervised and unsupervised learning, they employed polarity classification frameworks and support vector machines.

In 2020, Asif *et al.* [18] analyzed social media sentiment using multilingual text to assess extremist attitudes. A multilingual lexicon with different weights for intensity was constructed. The techniques were applied to Linear Support Vector Classifier and Multinomial Naive Bayes.

In 2016, Rojas-Barahona [19] stated that science and industry are more interested in automatically determining public polarisation on a topic. Social networks enable access to enormous blogs, suggestions, and reviews. The problem is to derive polarity from this data using opinion mining or sentiment analysis. This endeavour is complicated by subjective interpretation and linguistic processes that alter word polarity.

In 2021, Tam *et al.* [20] used the CNN model to extract text attributes and study local responses from the data of temporal or geographical. The computing complexity and training parameters were reduced in the CNN model using weight weight-sharing strategy. It cannot study correlation sequences, and its efficacy relies on window size

selection. RNN is a DL model that excels in sequential learning but struggles with parallel extraction of local features.

In 2020, Shahsavari *et al.* [21] described the current epidemic as a virus that spreads rumours and conspiracy theories, leading to real-world drama. The authors introduce machine learning methods for automated identification and identify narrative frameworks that encourage misleading propaganda. The study also found that misinformation and political ideology can harm public health.

In 2020, Yu *et al.* [22] used geographically situational awareness (GSA) in geospatial industries since SA-relevant information is largely spatial. GSA identifies damaged infrastructure, affected individuals, evacuation zones, and resource-poor communities. Social media's popularity and real-time data generation create new natural disaster chances. Social media has helped disaster management by communicating information during catastrophes.

In 2020, Cai *et al.* [23] explored the potential of deep learning-based multi-task learning models to address the challenge of limited training data. While these models significantly improved performance, some issues persisted. The addition of a pooling layer to the neural network of Bi-directional Recurrent (BRNN) to capture information of richer semantics, they attempted to get around these restrictions.

In 2019, Rehman *et al.* [24] highlighted the significance of sentiment analysis within the processing of Natural Language and machine learning. SA involves discerning subjective information in text and categorizing the writer's attitude as negative, positive, or neutral towards various entities. The quick growth of social media [25] has produced vast data, offering valuable insights into user opinions, emotions, and evaluations. Table 1 shows the analysis of the related works and research gaps.

A. Research Gap

Table 1: Analysis of the existing works

Author	Aim	Research gap
Kardakis <i>et al.</i> [16]	Explore how social media and the internet are amplifying sentiment analysis, underscoring its importance in deciphering user-generated content for businesses.	Examine the biases, ethical concerns, and language-cultural limitations of automated sentiment analysis. In-depth accuracy and limitation investigations can enhance research.
Song <i>et al.</i> [17]	Develop the model in an analysis of short text using PLTSs for word emotion polarity encoding and support vector machines for polarity classification.	Addressing scalability issues with large datasets, studying the model's pertinency across different topics and languages, and comparing it to other methods can improve its utility.
Asif <i>et al.</i> [18]	Conducting bilingual social media sentiment analysis for extreme sentiments using a sophisticated multilingual vocabulary and classifiers.	A comprehensive analysis should address biases, ethics, and compare with other methods to assess its performance in this area.
Rojas-Barahona [19]	The rising demand for sentiment analysis involves automatically uncovering public sentiments from social network data and opinion mining.	It could offer robust methods for handling context-based polarity shifts and improving sentiment analysis across various domains and languages.
Tam <i>et al.</i> [20]	To enhance CNN-based text attribute extraction from temporal or geographical data, utilizing weight sharing and integrating RNN capabilities.	Exploring novel approaches to address RNN gradient issues in CNN-RNN fusion. Improving interactions for enhanced information extraction.
Cai <i>et al.</i> [23]	To enhance sentiment classification, semantic information loss and shared feature selection must be overcome.	Researching the ASP-BRNN model's real-world effectiveness across various languages and domains. Assessing its adaptability to diverse natural language processing (NLP) tasks.
Rehman <i>et al.</i> [24]	Emphasizing SA in NLP and ML to extract subjective insights from web data across domains.	Addressing subtlety, irony, and cross-cultural text issues is vital. Research on precise, context-aware sentiment analysis, especially for multilingual data,

		is essential. Examining ethical issues like model prediction biases is crucial for real-world impact comprehension.
--	--	---

III. PROPOSED METHODOLOGY

The research proposal focuses on attractive sentiment analysis through EmotionNet, an innovative deep-learning framework. By participating in innovative ideas in preprocessing, feature extraction, selection, similarity score computation, and sentiment classification, EmotionNet endeavours are provocatively improving the accuracy and efficiency of sentiment analysis across various domains, including social media, customer reviews, and market sentiment. Figure. 1 depicts the overall proposed architecture.

A. Pre-Processing

In this study, preprocessing involves retaining advanced text-cleaning methods to eliminate noise, special characters, and extraneous data. This process employs neural text generation models, like transformer-based models, to normalize and cleanse the text data effectively.

1. Normalization

Text normalization is a central text preprocessing technique that involves normalizing textual data to ensure constancy and improve its suitability for NLP tasks. An incorporates various operations, such as lowercasing, eliminating punctuation, and stemming or lemmatization, aimed at reducing text variation and complexity. Standardization is the process of converting words into multiple forms. For shortening the data, the attraction of efficiency and minimizing the measurement of the algorithm of NLP has been used. Additionally, it is essential for addressing tasks like noisy text, inconsistent capitalization, and non-standard language. The application of the NLP facilitates precise and meaningful examination, classification, and sentiment investigation.

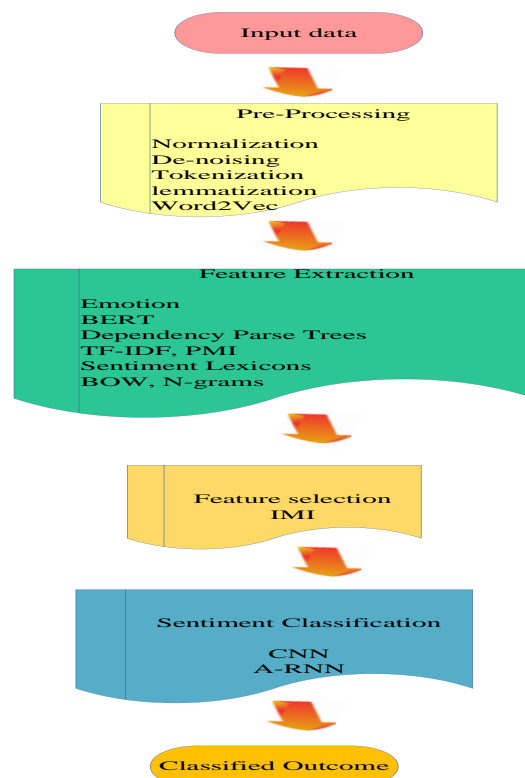


Figure 1: Overall Proposed Architecture

2. Denoising

When preprocessing data, it is important to consider text and images. Eliminating different categories of noise since textual input of HTML elements, special characters, and emojis, impair the precision and efficacy of an

activities of NLP. The neural text generation models are used to create pure noise-free text. The major ideas of an integrity text are proficient by positioning and eliminating unwanted noise. Similarly, photograph denoising deals with noisy images, which may arise from capture problems. The algorithm aims to eliminate noise from maintaining the image's important elements, image denoising returns photos to the quality of the original. It is crucial for appealing quality data, more precise investigation, and explanation of an applied text.

3. *Word Tokenization*

It is a natural language performance that breaks down text into reduced parts. Tokens represent words, sentences, and other important structures. The term the token is used as another way of approaching is division of the text into separate words and eradicating superior characters and punctuation. The approach is used for the processing of an efficient investigation of textual data. This method is used for different functions of evidence recovery, sentiment investigation, machine translation, and text organization. The main drawback of this method is the complicated language compound. For solving the drawback of an important scheme for organizing and controlling textual data investigation. The following categories of tokenization like subword, dictionary-based, and white space are used in this approach. The white token is used as divided into tokens of documents. Dictionary tokenization sections utilize pre-existing dictionaries. Subword tokenization reduces the text to smaller units.

4. *Lemmatization*

The NLP algorithm is used to minimize the number of words in a critical channel. To represent the lemmatization from running, ran, and runs, for instance. Lemmatization is an important tool in the NLP algorithm. The purpose of this method is to ensure spellings of the similar word are held properly by normalizing text data. The applications of this method are used as evidence retrieval, sentiment investigation, and text labelling. It helps chatbots and AI applications of the documents. Therefore, deciding among two techniques are accuracy and performance requirements.

5. *Word2Vec*

The representations of words are used in big datasets and neural networks. Word2Vec is a unique point in a multi-dimensional space to produce dense, continuous-valued word vectors. As a result, mathematical operations on word vectors can reveal analogies and semantic linkages since words that are semantically linked have vectors that show proximity. It is an essential tool for comprehending and interacting with human language because in different types of applications are used like sentiment investigation, machine translation, and document classification. An extraction of the feature stage is receiving pre-processed output.

B. *Feature Extraction*

Extraction of characteristics is used to analyze and understand text data's feelings and content. It covers a wide range of linguistic, semantic, and affective dimensions.

1. *Emotion Embeddings*

Conventional word embeddings are similar to the word2vec concentrate on representing words. It is an ignore of frequently an emotional subtlety of language, nevertheless. An essential for comprehending sentiment, affect, and the text's overall emotional tone. However, the term embeddings is used in an information of encoding emotional, and emotion embeddings. Some specific emotional categories are fear, rage, sadness, or happiness. An initial step of this embedding is the model of training models on emotion-annotated text data, and the particular technique employed produced models. Based on the investigation statistical approaches and neural networks are used.

2. *Contextual Embeddings*

By taking into account the larger context of words, contextual embeddings like BERT improve sentiment analysis and allow for a more sophisticated comprehension of the text's emotional content.

BERT

It's a bidirectional transformer. It demonstrates how to use the Bidirectional Encoder Representations from Transformers for NLP applications such as picture classification by including an image feature into an input. The complete process of classification utilizes an encoded information image and combines it with textual data.

Multiheaded Self-Attention (MHSA): An attention function represents the mapping from the query, collection of key-value pairs, and output. An output is calculated from the sum of the values with weight and the values were calculated from the function of query's compatibility. The mechanism according to Eq. (1) is the well-known scaled dot-product attention.

$$att(qu, ke, va) = softmax\left(\frac{quke^t}{\sqrt{d_{ke}}}\right) va \quad (1)$$

where d is an input data and $qu, ke, and va$ represents query, key, and value. It is written using p head (h_1, h_2, \dots, h_p) , w^o is matrices learning, as mentioned in Eq. (2).

$$mhsa(x) = concat(h_1, h_2, \dots, h_p)w^o \quad (2)$$

$$h_i = att(xw_i^q, xw_i^k, xw_i^v) \quad (3)$$

Concerning learned parameter matrices of $w^{qu} \in \mathbb{R}^{d \times d/p}$, $w^{ke} \in \mathbb{R}^{d \times d/p}$, $w^{va} \in \mathbb{R}^{d \times d/p}$ affine projections as mentioned in Eq. (3). The different heads of the MHSA mechanism each learn a distinct focus. Every head functions separately and simultaneously. Every attention distribution is computed using the scaled dot product attention.

3. Syntax-Based Features

Syntax-based features involve analyzing the structure and relationships between words in a sentence using techniques such as dependency parse trees.

Dependency Parse Trees

An understanding of the grammatical function and analysis of the text is used. The main purpose of this approach is to connect words from one to other words. Grammatical relationships of subject-verb links are revealed by the nature of these connections. An essential of these tasks are used in some research article acknowledgement, part-of-speech tagging, and machine translation. Because it allows an interaction of the among the words in the system of NLP. The major role of this method is an application of NLP. It is an improvement of machine learning and language understanding as well as understanding and interpretation of the NLP applications.

4. Semantic Features

Word co-occurrence patterns are revealed through the use of novel techniques such as TF-IDF and PMI to extract semantic characteristics.

TF-IDF

TF-IDF is a popular method for extracting relevant features from pre-processed text data. By calculating a score that accounts for both the word's frequency of the document and its frequency, the significance evaluates for the word within a document. The term score is a word w and document d has represented in Eq. (4).

$$TF - IDF(w, d) = TF(w, d) * IDF(w) \quad (4)$$

where $TF(w, d)$ is the frequency of the word w and document d , which measures how often appears the word in the document. IDF is an inverse of the frequency document word w , it measures word is the corpus of documents. An IDF score is a word w can be calculated as per Eq. (5).

$$IDF(w) = log(N / n_w) \quad (5)$$

where N represents the number of corpus documents and n_w represents number of corpus documents containing the word w . The term score gives a higher weight to words that appear frequently in a document. The words are likely to be more informative and relevant to the document content. After computing the TF-IDF scores for all words, the resulting vector of scores is used as features of downstream tasks.

PMI

PMI is a measure of statistics used to assess of association among random variables, X and Y, by comparing their joint distribution to their independent marginal distributions. The formula for calculating PMI is given as per Eq. (6).

$$pmi(f) = \log (p(f_{i,j}) / (p(f_i) * p(f_j))) \quad (6)$$

Where, $pmi(f)$ represents the PMI for the covariate pair, $p(f_{i,j})$ is joint probability distribution of the covariate pair for a specific time instant within the defined window W, $p(f_i)$ is the marginal probability distribution of all 24 covariates for a given hour, and $p(f_j)$ is the marginal probability distribution of a specific covariate for all 6 hours. The PMI value indicates the degree to which the observed relationship between two covariates (X and Y) deviates from what would be expected if they were independent. It quantifies how frequently temporal interactions occur between covariates in the specified time window, relative to their distributions. High PMI values (>1) suggest a frequent temporal interaction, while low values (<1) indicate a less frequent interaction compared to the marginal distribution. PMI values near 0 suggest interactions explained by the marginal distribution.

5. Sentiment Lexicons

Included are specific dictionaries that list words or phrases together with the accompanying sentiment polarities, which are often categorized as positive, negative, or neutral. It is controlling an emotional tone during the investigation of the tool. For instance, the positive words represent joy, whereas the negative words represent sad. It is a vital tool for an investigation of sentiment work since it computes growing sentiment scores, and enables context of a text. Confident files include words only, and polarity values, and other files are separated into files for positive and negative terms. The values of polarity are represented in ways, as real values in a range -1 to +1, as fixed positive or negative categorizations. It is an essential tool for sentiment investigation jobs to assist in categorizing text data's emotional tone and as a dynamic source of information for the model and algorithm of machine learning.

6. Textual Features

For the combination of word and frequency is text analysis, textual features like bag-of-words and n-grams record.

Bag-of-Words

Text data analysis and representation are done with it. This approach considers a document to be a bag of its individual words, ignoring context or word order in favour of word frequency. The BOW model works well in the following capacities: Text preprocessing, or tokenizing text into individual words, removing punctuation, and reducing text to small letters are among the first steps in the process.

To guarantee that the text is in a format that is consistent for analysis, follow these steps. To create a vocabulary, all of the unique words in the dataset or corpus must be gathered. Each word in the dictionary becomes a quality. Every document is converted into a vector of numbers. The vector's size corresponds to the vocabulary terms, and the values represent how frequently each word appears on the page. This makes a page look like a vector with word frequencies acting as its constituent parts. Numerous NLP activities can be completed once the text input has been transformed into these numerical representations. Textual data can be subjected to quantitative analysis and machine learning algorithms thanks to the BOW model.

N-grams

Natural language processing (NLP) employs the technique of extraction feature of N-gram of an extract word usage pattern since text. A n items of continuous sequence either word since particular text sample. It is a collection of multiple items that show up one after the other. Any written or spoken language can produce n-grams when it comes to natural language processing. The n-grams components represent words, sounds, characters, and the set of documents that occur frequently. It is represented based on the N-value, which denotes the sequence in the number of elements. A unigram, for instance, has an N of 1, a bigram or diagram, a trigram, and any number more than 3 is just termed an n-gram. It is the selection of N successive words and characters and omitting them once at a time.

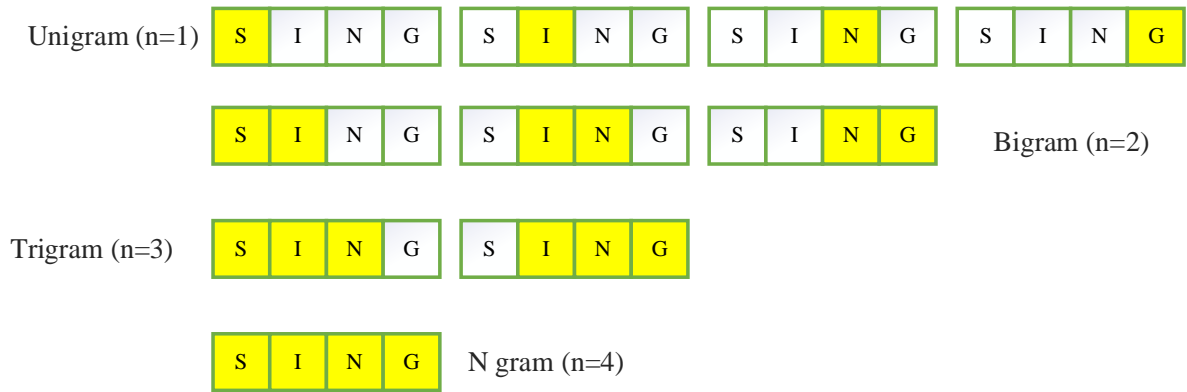


Figure 2: N-grams

The N-value directly correlates to improve the features produced using N-grams. The features can increase significantly as n gets bigger. The collected features are then passed on for additional processing during the feature selection phase, as seen above in Figure 2.

C. Feature Selection

In the feature selection phase, advanced techniques like IMI are employed to identify the informative features from an extracted set. This step enhances analysis accuracy by focusing on the most relevant aspects of the data.

1. IMI

MI is a crucial concept in information theory and statistics, measuring how much insight one random variable can provide about another. It quantifies the reduction in uncertainty regarding one variable when we possess knowledge of another. Mathematically, for two discrete variables *A* and *B*, it's expressed as per Eq. (7).

$$mi(A, B) = \sum_{b \in B} \sum_{a \in A} p(a, b) \log \frac{p(a, b)}{p(a)p(b)} * D(p(a)||q(a)) \tag{7}$$

Here, *p(a, b)* represents that the joint probability distribution of *A* and *B*, while *p(a)* and *p(b)* are their respective marginal probability distributions. The unit of measurement, often in bits, depends on the logarithm base used. MI possesses essential properties: it's always non-negative, symmetric, and additive for independent variables. The relative entropy, or Kullback-Leibler divergence, quantifies the difference between probability distributions *p(a)* and *q(a)*. When multiplied with mutual information, it reflects how much mutual information is lost or gained when transitioning from distribution *p* to *q*.

This combined metric helps assess information transfer and divergence between distributions. It also adheres to the data processing inequality, which means that processing information further downstream cannot increase its quantity. MI plays a pivotal role in communication channels. It helps determine the maximum number of messages that can be transmitted almost error-free and is closely tied to the concept of channel capacity.

D. Sentiment Classification

In sentiment classification, a novel deep learning architecture is proposed, which synergizes CNN, and A-RNN. This fusion aims to adeptly capture both fine-grained local and broader contextual sentiment information, enhancing classification accuracy.

1. CNN

It is a neural network of deep learning [26-29] designed to process ordered arrays of input, such as data. It is a feed-forward neural network of 20 or 30 layers. It is a powerful network due to the presence of a convolutional layer. Figure 3 shows the CNN architecture.

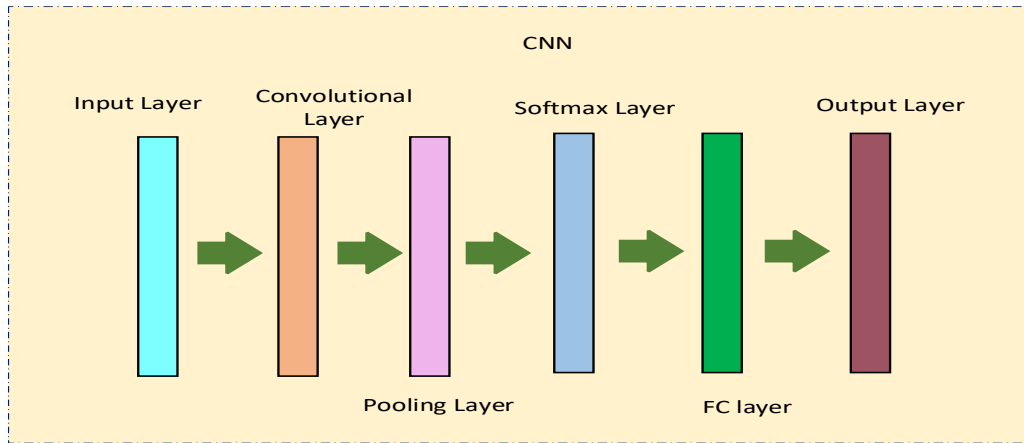


Figure 3: CNN model

Convolutional Layer

It is used to extract the different properties from input data. The operation of mathematical represents complications take place among input data and predetermined size filter $M \times M$. Descending the filter over an input data yields among filter and components of an input data based on the size of the filter's $M \times M$. An outcome has represented that the feature map offers specific data located at boundaries and corners. Subsequent layers of feature map to teach more features from incoming data. After application, the result is moved to the next layer by CNN's convolution layer. An operation of convolution is described as mentioned Eq. (8).

$$A_{i,j} = \sum_{q=0}^C \sum_{n=0}^C l_{q,n} s_{k+q,b+n} \tag{8}$$

Where $l_{q,n}$ represents convolutional kernel scale. $s_{k,b}$ represents an image of pixel value at k and b .

Pooling Layer

The purpose of these hidden layers in the model of CNN is to minimize the calculation to maintain the data and connect data sources. The major role of this layer is to minimize the size of the map and computing costs also. Each feature map is individually treated in this way, and the number of links between layers is decreased. Various types of pooling operations exist, contingent upon the chosen method. With the intention of gradually decreasing the feature map of the spatial dimensions while maintaining crucial material, pooling is frequently employed after each convolutional layer and given in Eq. (9).

$$MaxPool(e_o, u_o) = \begin{cases} e_o = floor\left(\frac{(e_i+2p-o)}{c} + 1\right) \\ u_o = floor\left(\frac{(u_i+2p-o)}{c} + 1\right) \end{cases} \tag{9}$$

Where, $Floor(Q)$ is the function of a number, e_o represents an output height, u_o represent an output width, e_i represents an intake height, u_i represents an input breadth, p represents the padding, o represents the size of the kernel, C represent the kernel stride.

Active Layer

It is used to determine the network of complex and continuous links among variables. Though used frequently and includes sigmoid, tanh, ReLU, and SoftMax functions, it determines the model of CNN whether or not to activate a neuron. This activation layer is used to predict the mathematical procedure.

Fully Connected Layer

The final layer of the CNN model is a fully connected layer, represented by the 1×1 finished layer, and $w \times h$ is used as global convolution, where w and h variables represent the width and height of the convolution layer.

2. A-RNN

It is a type of attention mechanism used in sequence-to-sequence models, particularly in tasks like machine translation. When creating the sequence of output, it enables a concentrated model on different segments.

Calculating the vector of context for each time step in the decoder while taking into account all of the encoder's concealed states is the fundamental principle of Bahdanau Attention. Figure. 4 depicts the architecture of A-RNN.

Producing the Encoder Hidden States: The encoder produces a set of hidden states, denoted as *Hencoder*, for each element in the input sequence given as per Eq. (10). These hidden states represent the information from the entire input sequence.

$$H_{encoder} = [h_1, h_2, h_3, \dots, h_n] \tag{10}$$

where *n* represents the length of an input sequence.

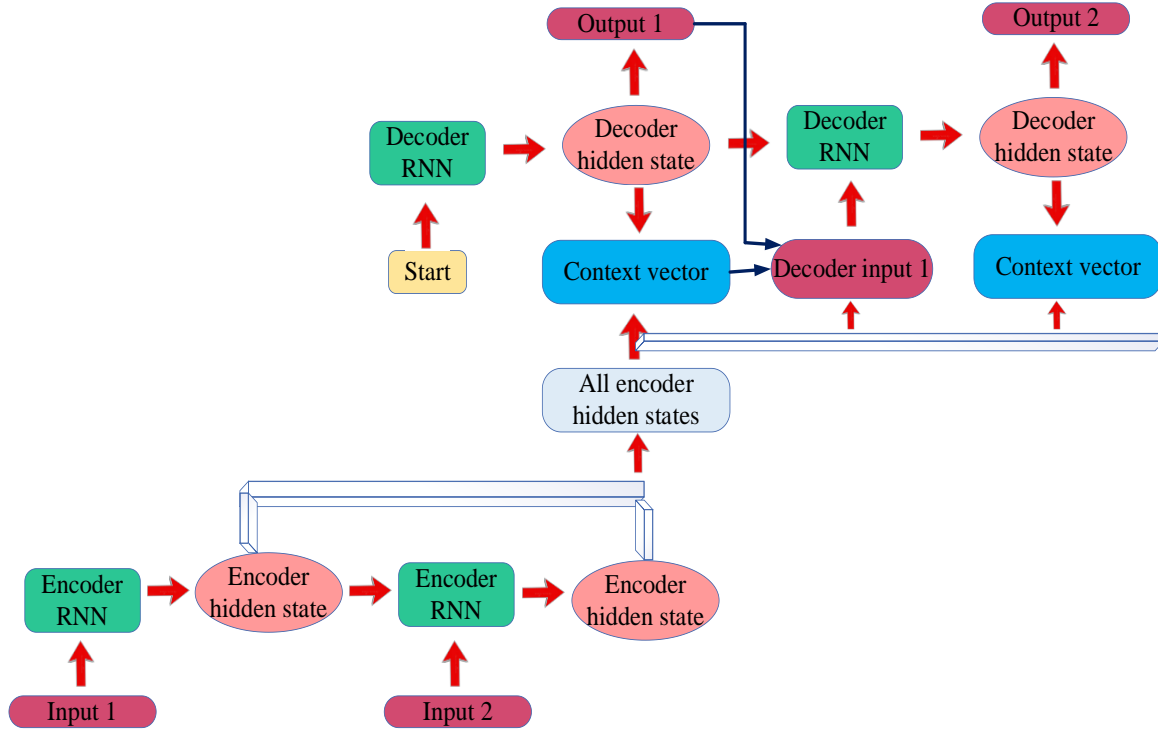


Figure 4: A-RNN

Calculating Alignment Scores: For each step time *t* in the decoder, you calculate an alignment score (or attention score) for each encoder hidden state. The alignment score ($e(t, i)$) for the *i*th encoder hidden state at the time *t* is calculated as mentioned in Eq. (11).

$$e(t, i) = a(s_{t-1}, h_i) \tag{11}$$

where s_{t-1} is the previous decoder's hidden state, and *a* is a learned function.

Softmaxing Alignment Scores: The alignment data is conceded the function to find weights consideration as ($\alpha(t, i)$). The sum of weights and attention is 1, probabilities and expressed as mentioned in Eq. (12).

$$\alpha(t, i) = \exp(e(t, i)) / \sum_j \exp(e(t, j)) \tag{12}$$

where *j* iterates over all hidden states' encoder.

Calculating the Context Vector: The vector (c_t) for the current decoder time step is calculated as the sum of weighted hidden states' encoder, with an attention weight, and it is expressed as per Eq. (13).

$$c_t = \sum_i \alpha(t, i) * h_i \tag{13}$$

where *i* iterates over all encoder hidden states.

Decoding Output: The vector is concatenated with an input of the present time step and used as input to decoder RNN, and it is calculated as mentioned in Eq. (14).

$$d_t = [y_{t-1}, c_t] \tag{14}$$

where y_{t-1} is the previous decoder output. RNN decoder generates an output for the current time step based on d_t . In the decoder the following above step for each time interval. Bahdanau mechanism allows to focus on decoder parts of the sequence an input for each generates output, quality of and improvement of the generated sequences. The parameters (function a) are learned during training. A popular loss function in neural networks represents the loss of cross-entropy, often known as categorical cross-entropy. The calculation of the actual label category and expected class probabilities. One way to represent the Eq. (15).

$$los = -\frac{1}{L} \sum_{i=1}^L = \sum_{j=1}^C y_{ij} \log(p_{ij}) \tag{15}$$

Where los represents the categorical cross-entropy loss, L represents the sample data, C represents the task of classes, i represents the sample of data an index ($i = 1$ to L), j is the index of the class ($j = 1$ to C), y_{ij} is a binary indicator (0 or 1) of whether class j is the correct classification for sample i , and p_{ij} is the predicted probability that sample i belongs to class j . In this loss function, for each data sample, the inner summation is taken over all classes (j) to compute the loss. The loss is higher when the predicted probability (p_{ij}) diverges from the true class label (y_{ij}), and it is minimized when the probabilities are predicted close to the true labels. By minimizing this loss, the multi-class classification model's accuracy is eventually increased by encouraging the model to assign higher probabilities to the correct class while lowering the probabilities for the wrong classes.

IV. RESULTS AND DISCUSSION

A. Experimental Setup

Python has been used to implement the suggested model. The evaluation dataset was gathered using the dataset [30]. The sentiment analysis using deep learning is elaborated in this section. The graphs are compared using several currently available methods, including gated recurrent units (GRU), recurrent neural networks (RNN), artificial neural networks (ANN), and convolutional neural networks (CNN). Performance indicators like accuracy, False Positive Rate (FPR), False Negative Rate (FNR), Matthews Correlation Coefficient (MCC), Negative Predictive Value (NPV), Correctness, Kindliness, Specificity, and F-measure.

B. Overall Performance Analysis of Existing and Proposed

Table 2 offers a thorough comparison of sentiment analysis performance metrics across several models, including the suggested model, GRU, ANN, CNN, and RNN. The models' overall correctness is gauged by accuracy. The proposed model achieves the highest accuracy at 95.79%, indicating that it correctly classifies sentiment with great precision. CNN and ANN also perform well in terms of accuracy, achieving 87.23% and 86.45%, respectively. The percentage of true positive forecasts to all positive predictions is quantified by precision. With a precision score of 90.99%, the suggested model shows promise in its capacity to provide accurate positive predictions. Recall, or the real positive rate, is represented by sensitivity.

The proposed model has achieved a sensitivity was 97.36% for positive. Similarly, the negative results of the proposed model have been achieved by 92.32%. For the measurement of the sensitivity and precision model was used as F-Measure. The proposed model has balanced precision and recall as a suggestion of F-measurement was 90.12%. MCC was used as the measurement of the classification of the binary quality. The categories of precision represented that an improvement performance was 92.33% in the proposed design compared to the MCC category. An NPV was a negative percentage of the genuine forecast. The maximum net current value was 93.43% of the prediction occurrence of the proposed model. The term negative fraction has mistakenly been mentioned as a positive term in FPR. For minimizing the positive errors of the FPR was 0.20% in the proposed design model. Similarly, the positive category of the fraction has mistakenly mentioned the negative category in FNR. The minimization of the negative error is represented as 0.20% in the proposed design model. The present investigation of the determination has represented an advantage of the analysis of sentiment.

Table 2: Overall Performance: Sentiment Analysis

Metrics	GRU	ANN	CNN	RNN	Proposed
Accuracy	0.73676	0.864545	0.872342	0.813423	0.957887
Precision	0.768787	0.756566	0.745646	0.809908	0.909879
Sensitivity	0.813435	0.812133	0.74232	0.86754	0.973553

Specificity	0.708767	0.834345	0.756754	0.797887	0.923243
F-Measure	0.79878	0.813334	0.776766	0.834454	0.901223
MCC	0.697878	0.797979	0.812323	0.771235	0.923342
NPV	0.809898	0.808123	0.754322	0.81233	0.934341
FPR	0.043212	0.040012	0.039865	0.031212	0.001976
FNR	0.002312	0.003112	0.004123	0.004678	0.001974

C. Overall Graphical Representation

Figure 5 represents that the complete listing of performance statistics and analysis which includes the metrics accuracy, F-measure, FNR, FPR, MCC, NPV, precision, sensitivity, and specificity. It's obvious from the graphs that the proposed model is better in all the performance metrics used.

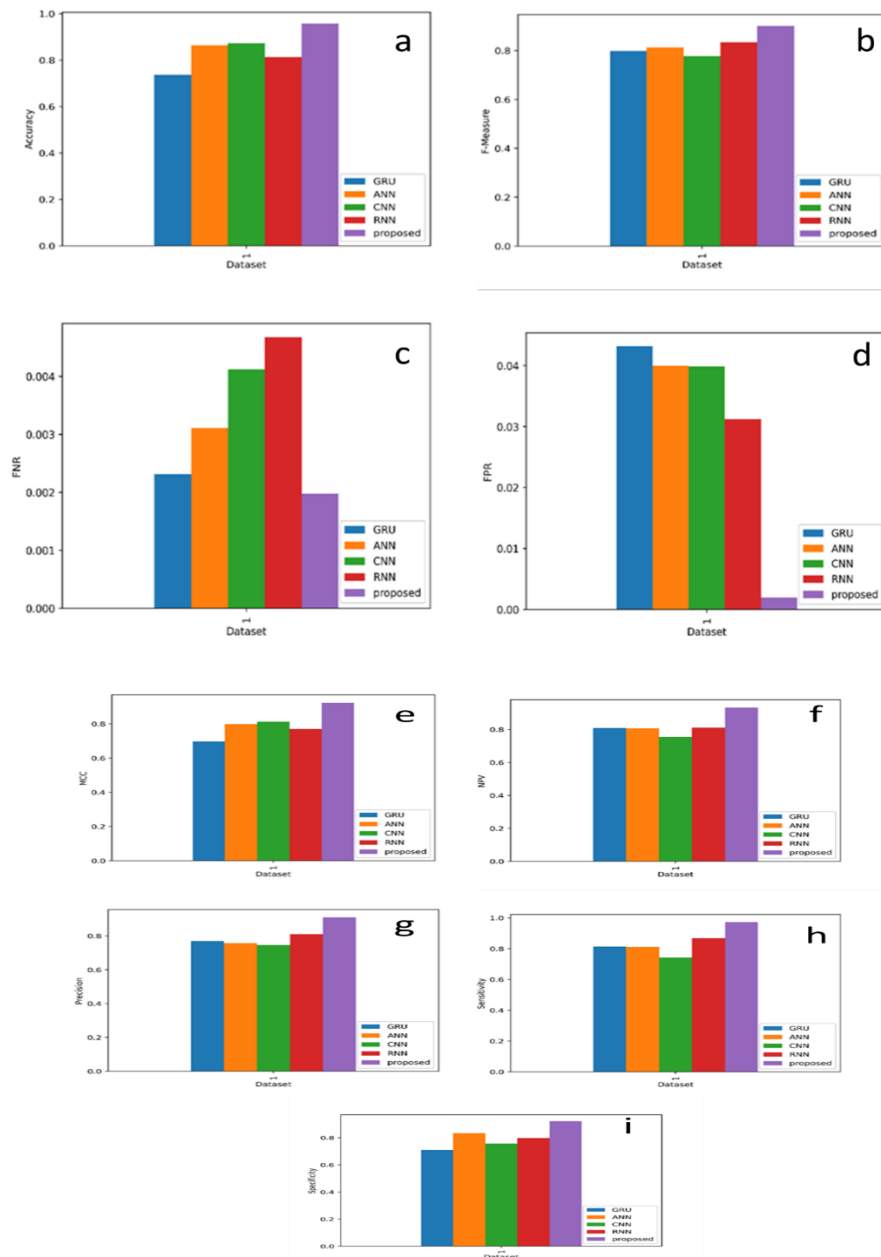


Figure 5: Overall performance analysis of a) Accuracy, b) F-Measure, c) FNR, d) FPR, e) MCC, f) NPV, g) Precision, h) Sensitivity and i) Specificity

V.CONCLUSIONS

EmotionNet, is a compound framework of deep learning methods projected to the growth of the accuracy and efficacy of an investigation in this research proposal. During the preprocessing phase, sophisticated performances for cleaning text were utilized. Elimination of noise from text was used as a transformer model. This was required for getting rid of superfluous symbols and standardizing text as non-standard. The text was further improved by lemmatization and tokenization, which reduced words to their most basic forms and divided them into meaningful units for deeper analysis. Word embeddings such as Word2Vec recorded semantic data. By extracting emotion and contextual embeddings, integrating syntactic features such as dependency parse trees, and putting forth unique semantic features based on TF-IDF and PMI to capture word co-occurrence patterns, feature extraction enhanced text analysis. Traditional textual elements and sentiment lexicons were combined as well. Feature Selection uses sophisticated methods, such as an IMI approach to choose the most useful features for sentiment analysis, to maximize efficiency. Sentiment classification developed an innovative deep-learning architecture combining CNN and A-RNN. The implementation tool used for the proposed model was Python. 95.7% accuracy was attained by the suggested model in the trials and assessments that were carried out.

REFERENCES

- [1] Kamath, U.; Liu, J.; Whitaker, J. Deep Learning for NLP and Speech Recognition; Springer: Cham, Switzerland, 2019; Volume 84. 5. Kwon, S. A CNN-assisted enhanced audio signal processing for speech emotion recognition. *Sensors* 2020, 20, 183.
- [2] Sajjad, M.; Kwon, S. Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM. *IEEE Access* 2020, 8, 79861–79875.
- [3] Paszkiel, S. Using neural networks for classification of the changes in the EEG signal based on facial expressions. In *Analysis and Classification of EEG Signals for Brain–Computer Interfaces*; Springer: Cham, Switzerland, 2020; pp. 41–69.
- [4] Aladem, M.; Rawashdeh, S.A. A single-stream segmentation and depth prediction CNN for autonomous driving. *IEEE Intell. Syst.* 2020
- [5] Kwon, S. Att-Net: Enhanced emotion recognition system using lightweight self-attention module. *Appl. Soft Comput.* 2021, 102, 107101.
- [6] Hossain, M.S.; Muhammad, G. Emotion recognition using deep learning approach from audio–visual emotional big data. *Inf. Fusion* 2019, 49, 69–78.
- [7] Anvarjon, T.; Kwon, S. Deep-net: A lightweight CNN-based speech emotion recognition system using deep frequency features. *Sensors* 2020, 20, 5212.
- [8] Pathak, A.R., Agarwal, B., Pandey, M. and Rautaray, S., 2020. Application of deep learning approaches for sentiment analysis. *Deep learning-based approaches for sentiment analysis*, pp.1-31.
- [9] Wei, J.; Liao, J.; Yang, Z.; Wang, S.; Zhao, Q. BiLSTM with multi-polarity orthogonal attention for implicit sentiment analysis. *Neurocomputing* 2020, 383, 165–173.
- [10] Dang, N.C.; Moreno-García, M.N.; De la Prieta, F. Sentiment analysis based on deep learning: A comparative study. *Electronics* 2020, 9, 483.
- [11] Yang, L., Li, Y., Wang, J. and Sherratt, R.S., 2020. Sentiment analysis for E-commerce product reviews in Chinese based on sentiment lexicon and deep learning. *IEEE access*, 8, pp.23522-23530.
- [12] Xu, D., Tian, Z., Lai, R., Kong, X., Tan, Z. and Shi, W., 2020. Deep learning-based emotion analysis of microblog texts. *Information Fusion*, 64, pp.1-11.
- [13] Ruz, G.A., Henríquez, P.A. and Mascareño, A., 2020. Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers. *Future Generation Computer Systems*, 106, pp.92-104.
- [14] Xu, G., Meng, Y., Qiu, X., Yu, Z. and Wu, X., 2019. Sentiment analysis of comment texts based on BiLSTM. *Ieee Access*, 7, pp.51522-51532.
- [15] Abdi, A., Shamsuddin, S.M., Hasan, S. and Piran, J., 2019. Deep learning-based sentiment classification of evaluative text based on multi-feature fusion. *Information Processing & Management*, 56(4), pp.1245-1259.
- [16] Kardakis, S., Perikos, I., Grivokostopoulou, F. and Hatzilygeroudis, I., 2021. Examining attention mechanisms in deep learning models for sentiment analysis. *Applied Sciences*, 11(9), p.3883.
- [17] C. Song, X. Wang, P. Cheng, J. Wang and L. Li, "SACPC: A framework based on probabilistic linguistic terms for short text sentiment analysis," *Knowledge-Based Systems*, Vols. 194, 105572, 2020. <https://doi.org/10.1016/j.knosys.2020.105572>
- [18] M. Asif, A. Ishtiaq, H. Ahmad, H. Aljuaid and J. Shah, "Sentiment analysis of extremism in social media from textual information," *Telematics and Informatics*, Vols. 48, 101345, 2020. <https://doi.org/10.1016/j.tele.2020.101345>

- [19] Rojas-Barahona, L.M., 2016. Deep learning for sentiment analysis. *Language and Linguistics Compass*, 10(12), pp.701-719.
- [20] Tam, S., Said, R.B. and Tanriöver, Ö.Ö., 2021. A ConvBiLSTM deep learning model-based approach for Twitter sentiment classification. *IEEE Access*, 9, pp.41283-41293.
- [21] Shahsavari, S.; Holur, P.; Tangherlini, T.R.; Roychowdhury, V. Conspiracy in the time of corona: Automatic detection of COVID-19 conspiracy theories in social media and the news. *J. Comput. Soc. Sci.* 2020, 3, 279–317. [CrossRef]
- [22] Salur, M.U. and Aydin, I., 2020. A novel hybrid deep learning model for sentiment classification. *IEEE Access*, 8, pp.58080-58093.
- [23] Cai, Y.; Huang, Q.; Lin, Z.; Xu, J.; Chen, Z.; Li, Q. Recurrent neural network with pooling operation and attention mechanism for sentiment analysis: A multi-task learning approach. *Knowl. Based Syst.* 2020, 203, 105856. [CrossRef]
- [24] Rehman, A.U., Malik, A.K., Raza, B. and Ali, W., 2019. A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis. *Multimedia Tools and Applications*, 78, pp.26597-26613.
- [25] Ahmad, S., Asghar, M.Z., Alotaibi, F.M. and Awan, I., 2019. Detection and classification of social media-based extremist affiliations using sentiment analysis techniques. *Human-centric Computing and Information Sciences*, 9, pp.1-23.
- [26] Almalis, I., Kouloumpis, E. and Vlahavas, I., 2022. Sector-level sentiment analysis with deep learning. *Knowledge-Based Systems*, 258, p.109954.
- [27] Alqarni, A. and Rahman, A., 2023. Arabic Tweets-based Sentiment Analysis to investigate the impact of COVID-19 in KSA: A deep learning approach. *Big Data and Cognitive Computing*, 7(1), p.16.
- [28] Vatambeti, R., Mantena, S.V., Kiran, K.V.D., Manohar, M. and Manjunath, C., 2023. Twitter sentiment analysis on online food services based on elephant herd optimization with hybrid deep learning technique. *Cluster Computing*, pp.1-17.
- [29] Ainapure, B.S., Pise, R.N., Reddy, P., Appasani, B., Srinivasulu, A., Khan, M.S. and Bizon, N., 2023. Sentiment Analysis of COVID-19 Tweets Using Deep Learning and Lexicon-Based Approaches. *Sustainability*, 15(3), p.2573.
- [30] Dataset taken from: “<https://www.kaggle.com/code/karan842/twitter-sentimental-analysis>”, dated 13/10/2023.