

An Insight Towards Prompt And Learning Based Sentiment Analysis And Classification With Feature Representation

¹Mr. M. Yuvaraja, ²Dr. C. Kumuthini

¹M.C.A., M.Phil., (Ph.d) Research Scholar Department of Computer Science Dr.N.G.P Arts and Science
College Coimbatore Tamilnadu India.
yuvaraja0105@gmail.com

² M.C.A., M. Phil., Ph.D. Professor, Department of Computer Applications Dr.N.G.P Arts and Science
College Coimbatore-641048 Tamilnadu, India. kumuthini@drngpasc.ac.in

Abstract- Sentiment analysis generally aims to ascertain a text's general sentiment polarity, which is a general strategy that doesn't particularly target any one target. On the other hand, aspect-based sentiment analysis has attracted a lot of interest lately. It is a more thorough approach that determines sentiment by looking at particular elements in the text. To accomplish this job, the majority of methods have historically relied on deep learning in conjunction with attention processes. DL-based models, however, can be intricate and need a lot of training. Learning-based devices have gained popularity recently due to their enhanced performance. Even if these models are successful, they are unable to distinguish between the multiple logical relationships among the data's characteristics and fail to accurately model them. This work suggests a hybrid Markov Long Short Term Memory (M-LSTM) method known as prompt-enhanced Sentiment Analysis in response to this. This technique employs the mask to represent mutual information between aspects and finds training data that is most comparable to the input text. Moreover, when conducting sentiment analysis on several aspects during inference, the suggested model only requires one forward pass. Additionally, the language representation model is utilized to improve the suggested M-LSTM method's performance. Extensive experiments were carried out to verify this strategy. Here, prompt-enhanced Sentiment Analysis, a novel and effective model, is presented. In this model, prompts from datasets that contain reciprocal information between aspects are merged. Cloze-style fine-tuning is used in prompt learning which functions similarly to pretraining. Consequently, it makes better use of pretraining data and greatly improves few-shot performance. To further enhance performance, the language representation model is also applied.

Keywords- sentiment analysis, prompt-based model, aspect model, information, training model.

1. Introduction

The fast growth of the World Wide Web, particularly social media, has produced a significant amount of textual data that contains people's thoughts and emotions. Even while this textual material is significant and may be used by businesses, the government, and other individuals to make decisions, an intelligent system that can automatically extract important information from it and categorize it according to its polarity needs to be developed. Sentiment analysis is one area of natural language processing that looks into this problem [1]. The advent of deep learning as a subset of machine learning has significantly advanced sentiment analysis in recent years, even though numerous studies have been carried out in recent decades to propose accurate methods for this task [2]. Computational techniques consisting of many processing layers can learn diverse data representations by taking into account the various levels of abstraction thanks to deep learning. Without

human assistance, deep learning techniques can use several processing layers to produce a variety of useful features from data [3], [4]. As a result, these techniques have significantly improved and advanced various domains, including natural language processing [7], computer vision [5], and speech recognition [6]. According to this trend, several deep learning approaches have been developed in recent decades, like recursive neural networks, CNN, and recurrent neural networks. Current research is focusing more on the possible uses and developments of these methods.

Online purchases of food, electronics, clothing, and other items have increased exponentially as a result of the COVID-19 threat. Customers have thus left a large number of internet reviews [8]. These reviews provide insightful information about the needs and satisfaction of the customer. The use of product/service-specific evaluations as a tool for analytical feedback analysis can boost the productivity of corporate sectors, such as restaurants and e-commerce websites. Additionally, a thorough analysis of competitors can be conducted using the reviews [9]. The speed, diversity, and volume of reviews that are frequently posted online, however, make analysis challenging. The lexicon-based method, hybrid methodology, and machine learning/deep learning-based method are the most commonly used techniques for sentiment analysis. The lexicon-based method determines the general tone of a sentence by using a vocabulary of sentiment-marked terms. To decrease the number of phrases that contain sarcasm, dependent clauses, or negations, sentiment ratings are occasionally combined with other criteria. The standards encompass a variety of NLP techniques, including stemming, tokenization part-of-speech (PoS), lexicons, and tagging. Lexicon-based sentiment analysis algorithms are thought to be limited since they do not account for the progressive integration of words [10]. Frequent fine-tuning and maintenance are necessary for Lexicon-based systems, which adds to the implementation effort. Machine learning or deep learning techniques are used to separate the dataset into testing and training datasets. Using a training data set, the model is trained to link the particular input text with the conforming outcome during the training phase [11]. A feature extractor is applied to convert hidden inputs in textual form into feature vectors, in the prediction phase, and the testing dataset is used. Prediction tags are produced for the matching vectors once these vectors have been fed into the model. An ML-based technique and a Lexicon-based strategy are combined in the hybrid method of sentiment analysis [12]. These methods show remarkable accomplishment in conventional text sources, structured domains where formerly labeled data is present or the lexical coverage includes terms that reflect particular moods inside a corpus. However, these methods cannot keep up with the vast amount, variety, and speed at which unstructured material is being uploaded to the internet. The development of numerous feature extraction techniques, which are necessary to generate reliable findings, has recently improved sentiment analysis methods' performance through machine learning techniques [13]. Finding unreliable and ambiguous characteristics for sentiment classification is one of the many difficulties that arise during the feature extraction phase. For a precise outcome that will get beyond these obstacles, a hybrid feature extraction model must be created. For accurate and intelligible feature extraction, particular products/services, their properties, and emoticons must be assessed in addition to sentence-specific or review-specific data. Look at the review "Although this mobile phone is too heavy, it is little cheap," which contains sentiment-conveying word connections and implicit elements like weight(as indicated by the term "heavy") and cost (as shown by the phrase "cheap") . Though elements have both positive and negative

polarity, sentiment generally seems neutral. Moreover, neither the definition nor the terminology used to represent these implicit elements are sufficient. This problem can be resolved by classifying highly comparable features into characteristics, after which the Aspect-based Sentiment Analysis (ABSA) method analyzes them [14]. Many different methodological approaches have been proposed in recent years; nonetheless, a more complex mechanism is required to manage ambiguities and unusual terms, recognize related characteristics, and record the relationship between implicit words. The two steps of the model's functionality—review mining and review summarization—are proposed in this research. In the review mining process, all reviews of the user are characterized as neutral, positive, or negative. The outcomes of the review mining procedure, review summarizing automatically creates a concise synopsis for related reviews. Developing a strong, reliable, and accurate sentiment analysis model, sometimes referred to as a review mining model, that can lead to the creation of a successful customer review summary model is the only objective of this endeavour [15]. This study addresses the issue of producing effectual review mining outcomes for customer review summarization by putting forth a novel prompt-based SA method founded on the reliable hybrid analysis of sentiments. First, stop words, numbers, nonsensical phrases, and other unwanted elements are removed using a quick and efficient pre-processing method. After that, the pre-processed assessments are fed into the dependable and effective hybrid M-LSTM feature engineering method and prompt-based Engineering, which associates aspects and review-related characteristics to produce an HFV. The HFV techniques presented in this work also manage inconsistent and unclear texts while doing sentiment analysis.

The work is organized as: section 2 depicts the functionality of diverse existing approaches. The methodology is drafted in section 3. The numerical outcomes are drafted in section 4 with summary in section 5.

2. Related works

The presented section contains of some existing investigations separated into groups such as ABSA, sentiment analysis, and DL-based SA methods [16]. The scheme suggested in this research is centered on the efficient analysis of sentiments. The purpose and significance of this study are also covered toward the conclusion of this section. To improve SA findings on Twitter data, SentiDif method [17] blends textual information with sentiment diffusion patterns. The authors first looked at a phenomenon called sentiment reversal to analyze sentiment diffusion, and they discovered several intriguing traits connected to sentiment reversals. Then, to forecast sentiment polarity from Twitter tweets, they took into account the correlations between the textual content in those messages and the sentiment diffusion patterns. Comparing the proposed approach to the advanced textual data-based SA algorithms, a wide range of experimentations on a real dataset express that it maximizes the region below the PR curve on the Twitter sentiment classification task. The author in [18] introduced a novel method to cross-domain emotion encoding challenges dubbed CrossWord, which uses the stochastic word embedding methodology. When paired with unlabeled reviews in both domains and labeled reviews in the source domain, the proposed method more reliably predicts probabilistic similarity correlations among pivot keywords and the terms in the source domain. The author in [19] announced a kernel optimization technique known as SentiVec. The second portion of this work uses an unsupervised updating model, such as object-word-to-surrounding-word reward models or context-to-object-word reward models, whereas the first part uses supervised learning. Based on

experimental results, the best sentiment vectors outperform existing methods on tasks like similarity, word analogies, and sentiment analysis effectively recovering the features in terms of sentiment analysis and semantics. The author in [20] use the BERT scheme to categorize public opinion on COVID-19. The authors used two data sets for sentiment analysis: one with tweets from people worldwide, and another with tweets from people in India. The experiment's findings demonstrate that 94% of sentiment classification is accurately validated.

The author introduced an innovative concept of the sentiment-based rating prediction approach [21] to construct a scheme that can extract useful data from social media user reviews and use ratings to identify the precise attributes that consumers will find appealing. This model uses a sentiment dictionary to calculate user opinions about an item. Moreover, the three feelings are used to compute item reputations to forecast and produce accurate recommendations [22]. A modern approach to semantic analysis and syntax, the n-gram methodology, increases the precision of the outcomes for the effective categorization of social media platform reviews when combined with SVM. To quantify sentiment, Ayyub et al. investigate several classifiers and feature sets [23]. The performance of ensemble-based approaches, cutting-edge DL-based methods, and conventional ML-based methods is empirically evaluated using the features set. The results indicate that classifier performance in sentiment quantification is influenced by different features. Furthermore, the outcomes show that deep learning methods achieve better than conventional ML processes [24]. Oyeboode and colleagues used five supervised ML techniques to carry out sentiment categorization on 88,125 reviews of hundred four mental health apps that were present on the Google Play and App Store. The classifier with the best performance was utilized to predict the review's sentiment polarity. Through a systematic analysis of both negative and positive evaluations, the researchers identified themes that indicate a range of characteristics that affect the mental health app's success in both optimistic and pessimistic ways.

In [25], the author offer a convincing paradigm to help close the knowledge difference between ML and lexicon-based methods. The scalability issue that comes up as feature collection grows is resolved by the authors' novel feature reduction method, which is based on evolutionary algorithms. The authors were able to cut the feature set's size by forty-two percent without compromising accuracy by using the hybrid strategy that was proposed. The author in [26] extracted attributes from online movie reviews and expressed those traits as a vector using the Bag of Words technique. Then, the authors used the Naive Bayes ML method for feature vectors to differentiate between positive and negative movie evaluations. Following categorization, the pairwise semantic resemblances among the review sentences were used to build an undirected weighted network [27]. The network edges indicated the weight of the semantic similarity, and the graph nodes represented the review sentences. For each review sentence in the network, the absolute measure was found through the weighted graph-based ranking algorithm. Finally, the highest-ranking metrics were used to identify which phrases (graph nodes) were the best, and these were used to build the extracted summary. Kumar et al. [28] introduced three sentiment analysis methodologies: converting processed corpora using Word2vec, mining opinions using convolutional neural networks (CNNs), and extracting semantic characteristics using ontologies. Non-dominant Pareto front optimum values for CNN parameter tweaking are found via particle swarm optimization. The author in [29] offers an aspect-based sentiment analysis methodology that uses sentiment extraction and polarity categorization on reviews

to automatically identify the most intriguing polarity traits that customers want. This approach blends deep learning and machine learning methods [30]. The goal of the search engine is to retrieve reviews and tweets on a user-specified term and present the relevant details.

3. Methodology

Based on the aforementioned problems with the state-of-the-art approaches, researchers suggest a novel SA strategy that focuses on consumer review summarization. The proposed approach consists of three steps: sentiment classification, feature extraction, and pre-processing. Within the pre-processing step, a common approach is used to denoise input reviews. Using a hybrid Markov Long Short Term Memory (M-LSTM) approach, the feature extraction stage cleverly addresses the challenges of effectively capturing the input review features. The projected paradigm specifically offers the below-given benefits: Because we found that hybrid feature sets yield useful outcomes, the feature extraction process was divided into 2 different fragments: Aspect-Related Features (ARF) and Review-Related Features (RRF). When the above approaches' numerical features are combined, a Hybrid Feature Vector (HFV) is produced.

Review-related features use a number of techniques to improve the accuracy of emotional representations, including n-gram features, emoticon polarities, and term frequency-inverse document frequency (TF-IDF). Our contribution to ARF is a novel method for assigning the polarities and extracting words with co-occurrence frequencies as attributes. The polarities of aspect words are produced via the ARF approach. An algorithm for deep learning classification. In the classification task, input reviews are divided into three categories: neutral, positive, and negative. Markov Long Short Term Memory (M-LSTM) is used for this work. The results of the previously described studies indicate that sentiment analysis has grown in prominence as a discipline of study because it may cover a lot of challenging and fascinating subjects. Because sentiment analysis has so many useful uses, a large number of start-up businesses are interested in using it. Every business is interested in learning how its clients see its goods and services. This field will continue to be active and vibrant in the years to come because of its functional components and scientific challenges. In an attempt to accurately execute sentiment analysis from the standpoint of summarizing customer reviews, a hybrid model is being developed. Fig 1 illustrates the many stages that were taken in this investigation. The first step includes gathering and pre-processing the data. ARF and RRF features are combined to create an HFV in the second stage, which involves feature extraction using both techniques. An algorithm for classifying data using BERT employs the hybrid feature vector to organize the input data as reviews into three groups: good, negative, and neutral. The model's M-LSTM performance is compared with the evaluation metrics of other models.

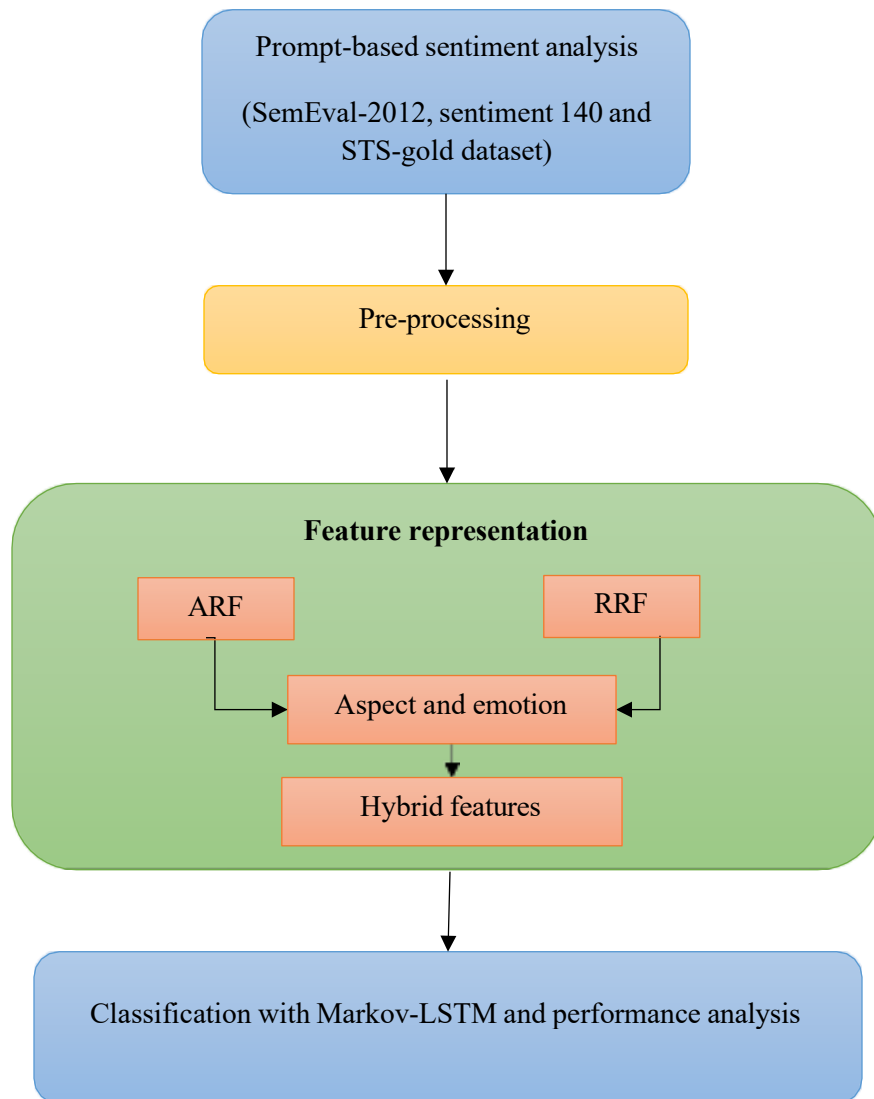


Fig 1: Flow diagram

3.1. Dataset

This work considers three diverse datasets like: SemEval-2012, Sentiment 140 and STS-gold datasets. The dataset is provided for pre-processing, feature extraction and prediction. The performance is evaluated with various metrics.

3.1.1. Pre-processing

To pre-process the dataset of raw reviews, the complex and unnecessary text is removed and fixed. Tokenization is the first step performed here, a feature utilized in the projected research, which concludes with the elimination of numbers, words with less than three characters, and meaningless terms. The input reviews were separated into tokens using the tokenization technique. After that, each token was stripped down to its most basic form via

a process called stemming. Then, stop-words like "I," "a," "an," "am," and so forth are eliminated to restrict the quantity of tokens. It finds and removes dates, URLs, special characters such as @, #, etc., and words that have no meaning, such as b+, C-, etc. Additionally, words and numbers with fewer than three letters are found and eliminated. This method ensures a significant reduction in the dimensions space of raw reviews.

3.2. Feature representation

Employing n –gram feature extraction to retrieve the list of words in the pre-processed reviews, resolved these problems. The TF-IDF of n-gram words is then computed with the help of term frequency-inverse document frequency on the n-gram yield. Together, TF-IDF and the n-gram methods reduce the dimensionality and provide each review with an effective representation. The features specific to emoticons are then retrieved to further maximize the sentiment analysis accuracy. Thus, RRF is produced by fusing emoticons-specific characteristics, n-gram, and TF-IDF. Below is a more thorough explanation of the RRF method. Presume that Q_i is the document that was pre-processed for the i^{th} review. Our first step is using the n –gram method on a pre-processed text source for TF-IDF and combined n-gram. The phrase "n-gram" describes a local data sequence of n-words. When $n = 1$, a model of an n-gram is termed a unigram; it is known as a bigram when n equals two, a trigram when n equals three, and so forth. The words "Bad" and "Very Bad," for instance, are bigrams and unigrams, respectively. According to the steps below, the n-gram technique uses the input sentence to generate a string of n consecutive words,

$$N_{gram} = \text{getNgram}(P_i, n) \quad (1)$$

Where, N_{gram}^i is the collection of n –grams obtained from the pre-processed input text Q_i . The parameters Q_i and n are transferred to the function $\text{am}^i N_{gram}^d$. When handling negations, they fixed the value of n to two to achieve a balance between economy and precision. The word list's IDF for extracted from the testing and training datasets is ascertained using TF-IDF after the n –gram results. While TF indicates how frequently a term occurs in a document, IDF determines how frequently a phrase appears across all of the papers in a corpus.

$$TF - IDF = TF(N_{gram}^i) * IDF(N_{gram}^d) \quad (2)$$

The stance adopted by N_{gram}^i signifies the word list for the i^{th} review and N_{gram}^d signifies the word list for the complete data set d. For instance, the TF value is 5 and $60 = 0.08$ if the word "good" occurs five times in N_{gram}^i 's 60 utterances. Similarly, if the word "good" appears 60 times in a 600-word sentence, the IDF is $\log(\frac{600}{60}) = 1$. The phrase "good" in the i^{th} review has a TF-IDF of $0.08 \times 1 = 0.08$ according to Eq. (2). Next, for every feature in the document, a vector $NT(i)$ is created:

$$NT(i) = N_{gram} + TF - IDF \quad (3)$$

All emoticon's unique components are then taken out of the review dataset and transformed into a vector for the process of classification. The emoticons feature vector's (EF) value is fixed which has a dimension of 1×2 , to zero for every review as reviews may or may not possess emoticons. In addition to the sentiment label, each review's emoticons are counted using the discrete probability distribution algorithm. A negative emoji is represented by a -1 and a positive emoji is represented by 1. The outcome will be displayed as $[3, -3]$ if the review has six emoticons, three of which are good and three of which are negative. Features connected to emoticons are assigned a value of zero if a review contains

neither positive nor negative emoticons or both. Eq. (3) is therefore rewritten as follows: NT traits are combined with emoticons-specific features.

$$\text{NTE}(i) = [\text{NT}, \text{EF}] \quad (4)$$

3.3. Classification of Sentiments using LSTM

The development of the suggested hybrid Markov Long Short Term Memory (M-LSTM) methodology for indoor location prediction is depicted in Fig. 2. The bottom-up design principle serves as the foundation for the four elements of our methodology detecting location sequences, constructing a multi-step transition probability matrix, fusing various Markov chains, and finding neighbouring locations. First, the trajectory cannot be directly included in the prediction model due to the continuity of space. Therefore, researchers need to transform the route into a location sequence linked to a certain store. A specific multi-step Markov transition probability matrix is constructed to convert a Markov chain in higher order into many first-order Markov chains. Then choose the nearby locations that are best for each user. These first-order Markov chains are then integrated using the Markov-LSTM model to produce the target user's anticipated outcomes.

3.4. Sequence analysis

It is more likely that the application service information will be viewed when the user is expressing the emotion. The emotion point was extracted from the individual trajectory in this work using the approach. The individual trajectory is divided into k different groups $\{C_1, C_2, \dots, C_k\}$, each with k stay points using the proposed model. The existing model is an improved algorithm designed to reinterpret the inside space's spatiotemporal neighbourhood. The trajectory point and its spatio-temporal neighbourhood is expressed through the formula below:

$$N_{\epsilon_1, \epsilon_2}(pt_i) = \{pt_j \in \text{traj} | \text{sd}(pt_i, pt_j) \leq \epsilon_1 \text{ and } \text{td}(pt_i, pt_j) \leq \epsilon_2 \text{ and } pt_j, f_j = pt_i, f_i\} \quad (5)$$

Here, a function that determines the spatial distance among pt_i and pt_j is represented as s_d to estimate the time distance between pt_i and pt_j we used td and $N_{\epsilon_1, \epsilon_2}(pt_i)$ is a set of points that contain the spatiotemporal neighborhood. Only spatial information, i.e. not semantic information is present in the stay point that the proposed model discovered. To give each stay point semantics, researchers employ the NN search. When it is inside the store, it uses the intersection approach to identify which store user you visited at stay point. The shop closest to a stay point 2 outside the shop is identified together with the appropriate distance d . The user visits at stay point is the nearest one if d is less than the distance condition δ .

3.5. Probability analysis

A traditional statistical model known as the k – MC (matrix) uses transition probability to explain how a user moves between sites. The k – MC consider user location sequence's all location as a random variable and to estimate the future position of a user, it makes use of joint probability. An example would be to utilize user u as a random variable to generate a location sequence of length $\{l^u\}_m^1$. The user u 's k – order probability transition matrix can also be expressed as $P(L_{m+1}^u | L_m^u, L_{m-1}^u, \dots, L_{m-k+1}^u)$, which is $Y^u \in R^{k \times N \times N}$. As k rises, it becomes less feasible to employ k – MC for position prediction because the transition probability matrix, Y^u of user u rapidly rises. This suggests a new Markov chain with k steps, represented by $\text{MC}^{(k)}$.

Given that $Y^{u(1)}$, the user's 1-step transition probability matrix equals $Y^{u(1)}$, the 1-order transition probability matrix, it is likely that user u will move from position l_i to location l_j

in a single step. This is its fifth probability matrix definition. The definition of $Y_{ij}^{u(1)}$ is given by the following phrase:

$$Y_{ij}^{u(1)} = \frac{\sum_{p=1}^{m-1} |\{l_p^u = l_i \cap l_{p+1}^u = l_j\}|}{\sum_{p=1}^m |\{l_p^u = l_i\}|} \cdot \frac{Y^{u(1)} \in R^{N \times N}}{l_p^u \in \text{locSeq}^u} \quad (6)$$

The overall emotion by user u from location l_i to other locations in a single step is $\{l_i^u\}_{i=1}^m$ and the total number of sentiment is N . The emotion expressed by user u from location l_i to location l_j in a single step is $\sum_{p=1}^{m-1} |\{l_p^u = l_i \cap l_{p+1}^u = l_j\}|$. The location sequence is locSeq^u and $\{l_i^u\}_{i=1}^m$ of user u .

Probability Matrix for k -Step Transition: The $N \times N$ matrix $\hat{Y}^{u(l_i \rightarrow *:k)} = Y^{u(k) \rightarrow *}$ signifies the k -step transition probability matrix of user u . It shows the likelihood that user u will express from point l_i to various destinations in k steps. The definitions of $Y^{u(k)}$ and $\hat{y}^{u(l_i \rightarrow *:k)}$ for user u are given in Eq. (7) and Eq. (8) respectively:

$$Y^{u(k)} = P(L_{m+1}^u | L_{m-k+1}^u) Y^{u(k)} \in R^{N \times N} \quad (7)$$

$$\hat{y}^{u(l_i \rightarrow *:k)} = P(L_{m+1}^u | L_{m-k+1}^u = l_{m-k+1}^u) \hat{y}^{u(l_i \rightarrow *:k)} \in R^{1 \times N} \quad (8)$$

where, $L_{m-k+1}^u = l_{m-k+1}^u$ indicates that user u has chosen for emotion of l_{m-k+1}^u at random variable and $Y^{u(k)}$ can be computed from $Y^{u(1)}$ directly so that $Y^{u(k)} = Y^{u(1) \cdot k}$. $Y^{u(k)}$ explains how cross-location impacts the prediction results from a different angle, L_{m-k+1}^u (which may be derived from the position sequence) and the number of stores is given as N . If $MC^{(k)}$ is present, the size of the transition probability matrix that corresponds to the $1 - MC$ transition probability matrix will also alter. The $k - MC$ can be divided into k first-order MC s using this matrix which lowers the transition probability matrix's dimensions and eliminates the need to compute $k - MC$'s joint probability. This theoretical analysis with mathematical evidence of the probability matrix $Y^{u(k)}$ for k -step transitions. If the user u 's position at random variable L_{m-k}^u is L_{m-k}^u , then the following formulae can be used to define the likelihood that user u will move from place L_{m-k}^u to other places in k steps:

$$\hat{y}^{u(l_i \rightarrow *:k)} = Y^{u(1)} \quad (9)$$

Here, $P(L_{m-k+2}^u | L_{m-k+1}^u)$ is the formula, where $P(L_{m-k+2}^u | L_{m-k+1}^u) * P(L_{m-k+3}^u | L_{m-k+2}^u)$ are two possible representations of the k -step transition probability matrix $Y^{u(k)}$. One way to express $Y^{u(1)}$ is as $(Y^{u(1)})^k$, or $Y^{u(k)} = (Y^{u(1)})^k$, since matrix multiplication satisfies the associative condition.

3.6. Finest analysis

As in the $k - MC$, the Markov-LSTM model as well needs to figure out the hyper-parameter, k , or how many places affect the prediction's outcome. The effectiveness of the parameters is a major factor in the model's prediction abilities. Determining the number of nearby sites is the main goal of the k value. The model represents a first-order Markov chain if k is too small which lowers the prediction performance. Excessive k -parameter increases might cause overfitting and model confusion. Cross-validation is widely employed to determine the k value that diminishes the prediction error of the model since it has a significant effect on prediction performance. Because each user in this work is an independent person,

researchers choose the best k value for each user. When $k_u > 1$, the k -MC can be computed using the below-given expressions, which use a k value of user u as k_u :

$$\begin{aligned} \hat{y}^{u(l_i \rightarrow *:k)} &= P(L_{m+1}^u | L_m^u = l_m^u) \\ \{ \hat{y}^{u(l_{m-k_u+1}^u \rightarrow *:k)} \}_{i=1}^{k_u} &= P(L_{m+1}^u | L_{m-k_u+1}^u = l_{m-k_u+1}^u) \end{aligned} \quad (10)$$

where, $\{ \hat{y}^{u(l_{m-k_u+1}^u \rightarrow *:k)} \}_{i=1}^{k_u}$ denotes the outcomes of several first-order Markov models' predictions.

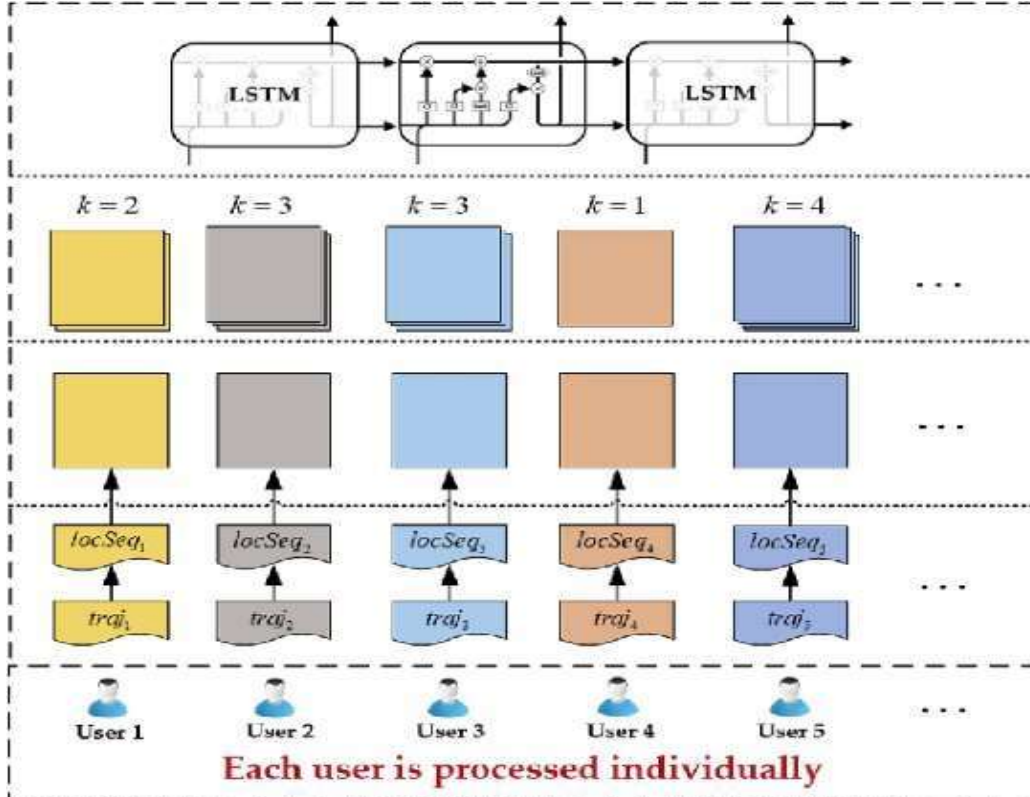


Fig 2: Proposed model

3.7. Fusion model

S- First-order Markov models of size k_u are established for every user, u . However, each model's capacity to predict the subsequent position is restricted. To guarantee strong location prediction performance, this study integrates k_u first-order Markov models. While taking into consideration the order of the k_u first-order Markov model prediction outputs,

which is $\{ \hat{y}^{u(l_{m-k_u+1}^u \rightarrow *:k)} \}_{i=1}^{k_u}$, they integrate the k_u discoveries using the LSTM model. Two

factors can be taken into consideration when looking for ways to improve model prediction performance: (1) From the standpoint of an LSTM model, the multi-step transition probability matrix avoids dimensional disasters by permitting the application of multiple k -MCs; (2) instead of mining the simple identification sequence's location pattern, our model extracts the transition probability's location pattern which comprises additional

movement tendencies of a person. To effectively control the rate of information accumulation, the LSTM model, the RNN model's extension, implements a specific cell with gate mechanisms and forgets specific historical accumulation of information selectively. The outputs of first-order Markov models are progressively merged using the input gate, forgetting gate, and output gate, as shown in Fig 2. This fusion approach additionally considers the independent influence of several outputs on the prediction results to determine the number of outputs that interact. The user u's fusion process can be designed using equations such as these:

$$f^m = \sigma(W_{hf}h^{m-1} + W_{yf}y^{u(lm \rightarrow *:1)} + b_f) \quad (11)$$

$$i^m = \sigma(W_{hi}h^{m-1} + W_{yi}y^{u(lm \rightarrow *:1)} + b_i) \quad (12)$$

$$a^m = \tanh(W_{ha}h^{m-1} + W_{ya}y^{u(lm \rightarrow *:1)} + b_a) \quad (13)$$

$$C^m = C^{m-1} \odot f^m + i^m \odot a^m \quad (14)$$

$$o^m = \sigma(W_{ho}h^{m-1} + W_{oy}y^{u(lm \rightarrow *:1)} + b_o) \quad (15)$$

$$h^m = o^m \odot \tanh(C^m) \quad (16)$$

$$\hat{y}^m = \sigma(W_h h^m + b_h) \quad (17)$$

Here, W_{hf} , W_{yf} , W_{hi} , W_{yi} , W_{ha} , W_{ya} , W_{ho} , W_{oy} , and W_h are the weight matrices; $\hat{y}^m$ is the Hadamard operation; y^m is the Markov-LSTM model's output, or the problem definition is C^m , i^m , o^m and f^m etc. are the forgetting gate, input gate, control unit, and output gate; σ is the sigmoid activation function. The hidden unit of correlation among the outputs of various Markov models is h^{m-1} . By merging multiple markov outputs, we can train our markov-LSTM to predict $\hat{y}^m$ while lowering the cross-entropy loss among a user's true and projected locations. It is possible to define this loss function using the following equation.

$$L(\theta) = - \sum_{j=1}^N y_j^m * \log(\hat{y}_j^m) \quad (18)$$

Here, N is the total number of emotions; the output of the model is y_j^m ; and the model's true value (projected output) is $\hat{y}_j^m$. Each learnable parameters are θ , or all b and W parameters in the Markov-LSTM model. The three components of the user u location sequence, $locSeq_u$ that are utilized in this investigation are test samples, training samples, and historical samples. From the historical location sequence, $Y^{u(k)}$ provides the user with u 's k -step transition probability matrix. The model θ parameter is trained using the training location sequence, and the model's prediction performance is assessed using the test location sequence. Mu's training procedure is shown in Algorithm 1.

3.8. Complexity analysis

In the Markov-LSTM model, determining the k -step Markov model is an important process. The benefits of the k -step markov model are examined in this section from the standpoints of computational and spatial complexity. From the space complexity's perspective, the number is $k * N * N$, even though the k -step transition probability matrix has $N * N$ entries. This matrix uses a lot less storage space than the k -order Markov transition probability matrix. The benefits of the k -step Markov model increase in importance as k increases. The computation of the first-order Markov transition probability matrix is the only computational expense associated with the k -step Markov model as demonstrated. When compared to the k -order Markov model the k -order Markov transition probability matrixes processing cost is more. The k -order Markov transition

probability matrix is harder to produce as k rises. Even with the increase in k , there is no positive growth in the k –step Markov matrix of transition probability

4. Numerical results and discussion

4.1. Tools and analysis

Researchers employed an i5 processor running Windows 11 and the MATLAB application to conduct an investigative examination of the suggested job. Researchers evaluated the projected methodology and contrasted it with the latest advanced approaches using Sentiment140, STS-Gold, and the SemEval-2014 dataset containing restaurant reviews. 30% of each of the three datasets are used for testing, and 70% are used for training. By contrasting the HFV performance with the recommended RRF and ARF approaches on the SemEval-2014 restaurant reviews dataset using the LSTM classifier, the efficiency of the hybrid feature engineering approach is illustrated. The proposed hybrid feature extraction method is also compared with the latest innovative method's outcomes. Several criteria, such as accuracy, F1-measure, recall, precision, and MSAT, are utilized to evaluate the performance of, RRF, HFV, and ARF. The MSAT is a measure of the average processing time, or the amount of time required to calculate the sentiment categorization. Researchers performed 50 sentiment analysis classification cases to ascertain the best way to estimate the assessment parameter.

4.2. Performance evaluation

This section uses the SemEval-2014 restaurant ratings dataset to evaluate the ARF, HFV, and RRF performance through M-LSTM. Various evaluation metric's outcomes are presented in Figs. 3 and Table 1 and Table 4 displays the related results. The hybrid feature engineering approach outperformed both ARF and RRF utilizing M-LSTM in sentiment classification, according to an analysis of accuracy, F1-measure, recall, and precision. The M-LSTM technique which handles sarcasm, denial, uncertainty, feelings, attitudes, and opinions surrounding restaurants and their attributes, is the main reason for this performance improvement. TF-IDF, n-gram, and emoticons-specific features are combined to generate the RRF vector. Effective construction is used to create the hybrid feature vector, which contains information particular to emoticons and the frequency of phrases that appear in each review. The ARF technique handles ambiguous and caustic phrases by using the co-occurrence frequency strategy, which further reduces sentiment analysis errors. As a result of its ability to effectively handle opinions, emotions, and negations, the RRF fared better than the ARF. The hybrid M-LSTM model's performance and F1-measure with varying sizes of training data sets are illustrated in Fig 4, respectively. Stratified sampling is used to split the training dataset into minor clusters of variable sizes. Training data increases by 10% with each iteration, as shown in Figs 5 and Fig 6. The F1-measure performance of the HFV approach is superior to that of the ARF and RRF approaches. Because more labelled data is there for exact prediction, the F1-measure performance rises. The M-LSTM analysis indicates that ARF and RRF feature extraction algorithms require less processing time than HFV. However, it makes sense considering the advancements the HFV has made in sentiment analysis.

Table 1: Feature extraction with proposed M-LSTM

Features	F1-score	Precision	Recall	Accuracy
ARF	89	90	87	89

RRF	92	94	92	92
HFV	96	98	96	95

Table 2: Comparison with SemEval-2014 dataset

Approaches	Precision	Recall	F1-score	Accuracy	AUC
SABSA	84	83	83	83	82
SentiVec	86	84	85	85	83
N-gram	85	83	84	84	82
SEML	84	82	83	82	81
MVN	79	77	78	77	76
LSTM	94	91	92	90	91
M-LSTM	96	96.5	97.1	92	93

Table 3: Comparison with Sentiment140 dataset

Approaches	Precision	Recall	F1-score	Accuracy	AUC
SABSA	85	83	84	83	83
SentiVec	87	85	86	86.1	85
N-gram	86	86	85	85	83
SEML	85	83	84	83	82
MVN	81	78	80	79	78
LSTM	95	93	94	92.1	93
M-LSTM	97.1	97.5	96.1	95	95.5

Table 4: Comparison with STS-gold dataset

Approaches	Precision	Recall	F1-score	Accuracy	AUC
SABSA	79	78.5	78.7	78	76
SentiVec	84	83.5	85	85	81
N-gram	83	81.7	82.4	81	80
SEML	83	81	81.7	82	79
MVN	78	75.8	76.5	76	74
LSTM	92	90	91	90	90
M-LSTM	95.7	95	94	95	96

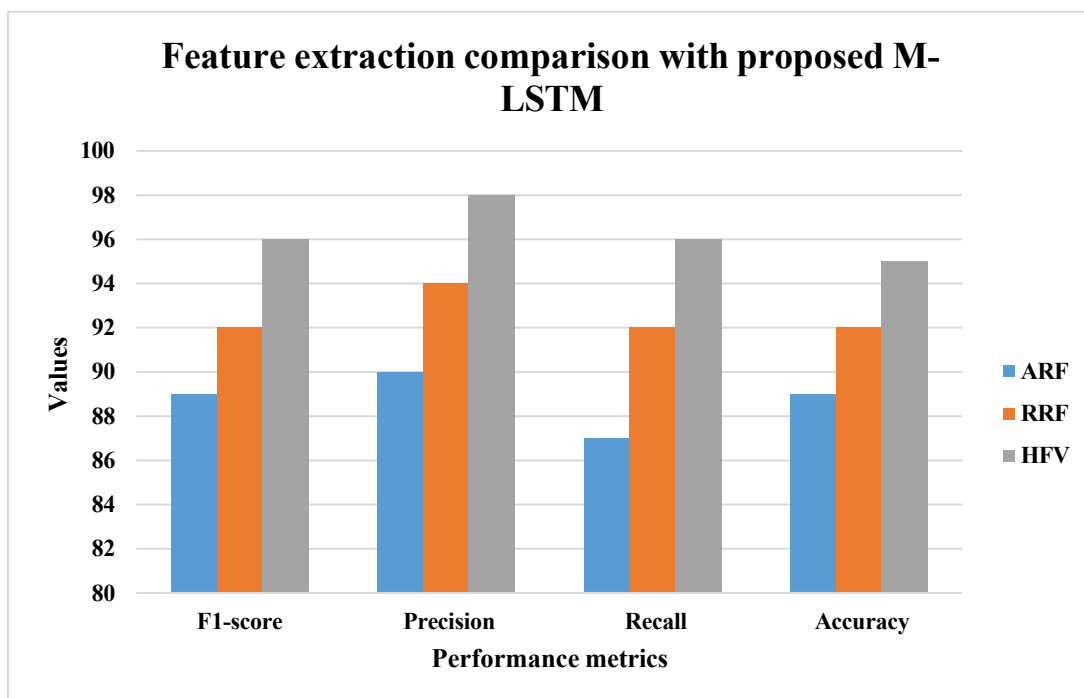


Fig 3: Feature extraction comparison with proposed M-LSTM

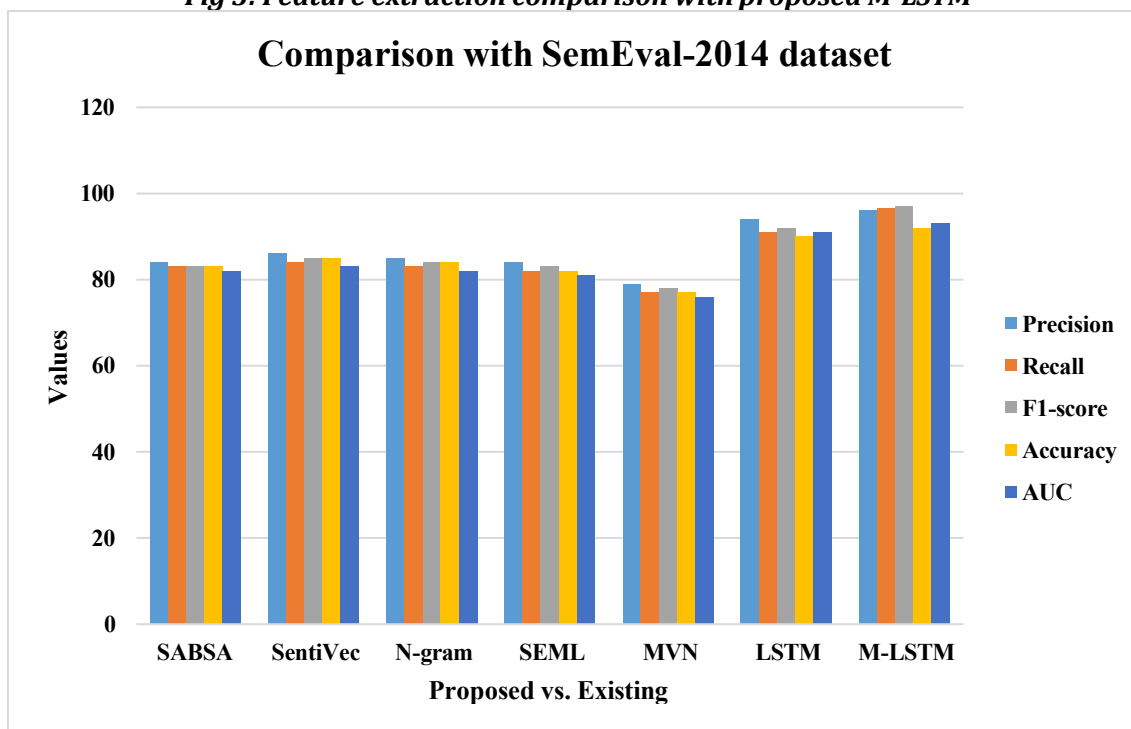


Fig 4: Comparison with SemEval-2014 dataset

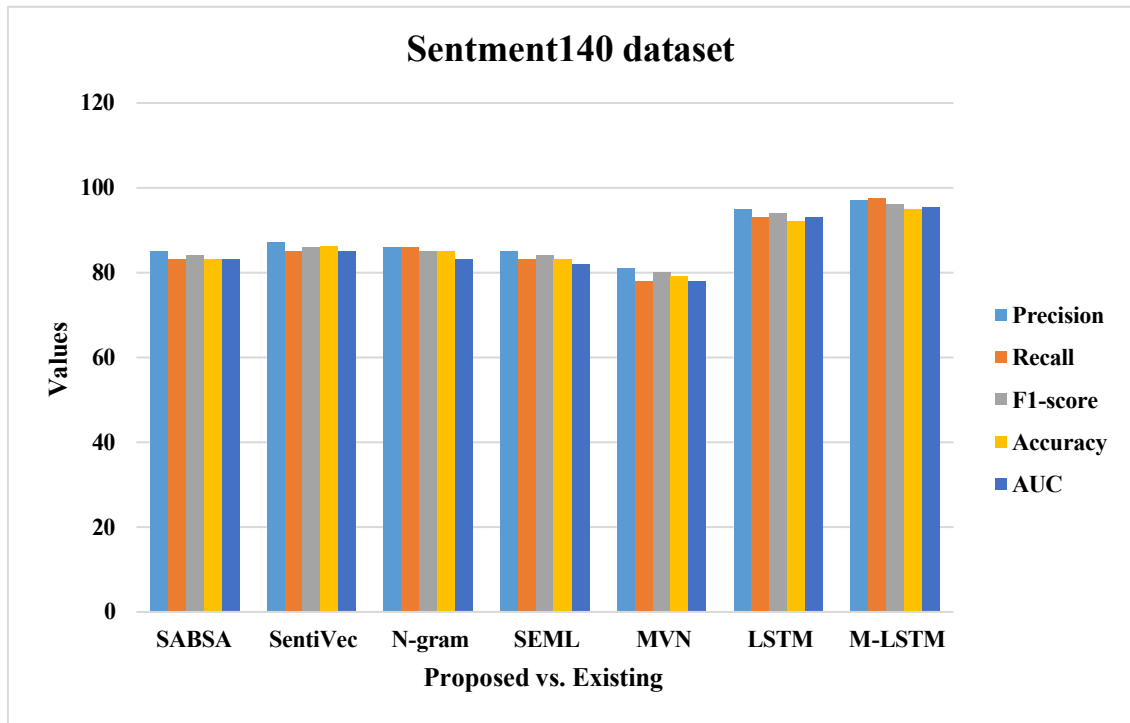


Fig 5: Comparison with Sentiment140 dataset

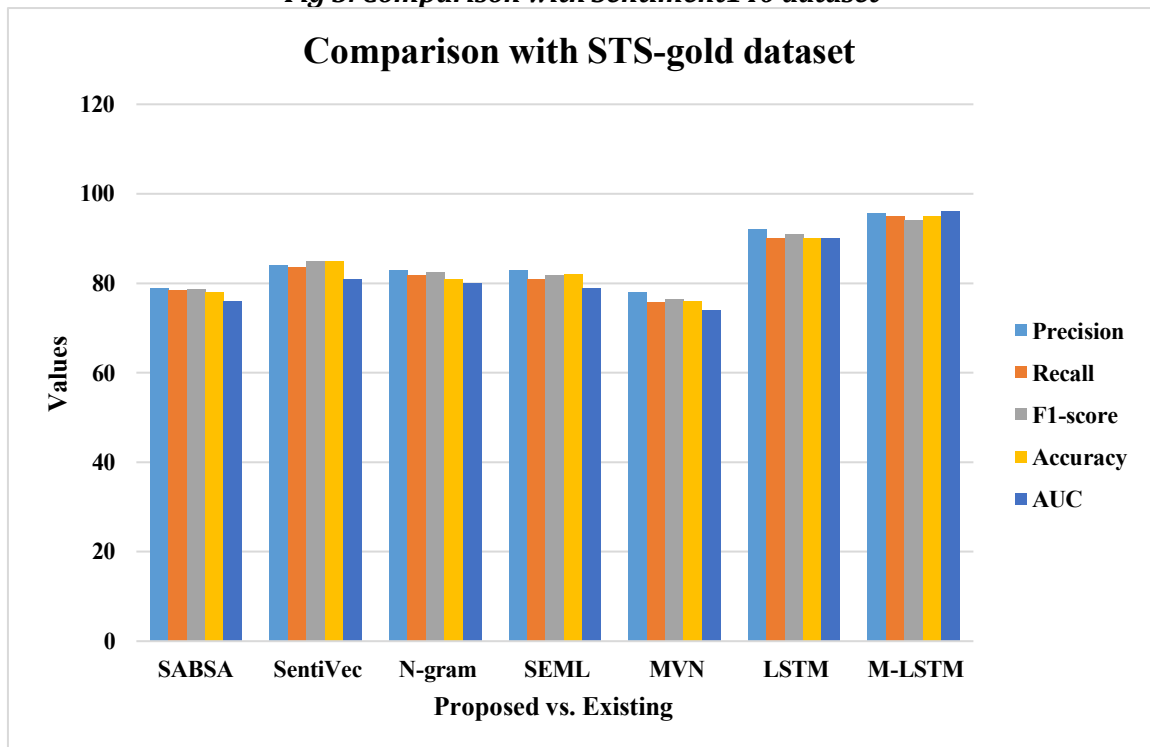


Fig 6: Comparison with STS-gold dataset

Table 3 evaluate all of the previously described methods using the SemEval-2014 restaurant review dataset, and Sentiment140 dataset according to and STS-Gold dataset to recall, F1-measure, precision, accuracy, and AUC evaluation criteria. For each analytic parameter, the hybrid feature extraction method utilizing M-LSTM outperforms the other methods in all three datasets. Its capacity to reliably and effectively construct the hybrid feature vector utilizing a variety of ways is what leads to enhanced performance. Sentiment140 performed better than SemEval-2014 and STS-Gold in terms of precision, accuracy, F1-score, recall, and AUC outperforming the two datasets. The Sentiment140 dataset has over 1,60,000 reviews, which is significantly more than the other two datasets. This is the reason for the improvement. Further precise classifications using each classifier are possible since each class in the dataset comprises a large number of examples. STS-Gold did worse across all classifiers since it had fewer training examples than the Sentiment140 and SemEval-2014 datasets. Because N-gram+TF-IDF+SVM and SentiVec lack negation handling and are unable to extract review-specific properties, they perform better than the ABSA techniques, MTMVN SEML and SABSA. To enhance the result, the model proposed in this study provides a combined solution that integrates the benefits of RSF and ASF.

5. Conclusion

The presented work's main objective is to precisely assess sentiment from the standpoint of the prompt-based emotion recognition of customer review with prompt-engineering model. To address the difficulties that arise while conducting sentiment analysis on unfiltered emotion-based reviews, we listed several research issues and potential fixes. This work proposes a hybrid feature extraction technique that may transform the pre-processed reviews into relevant feature vectors allowing for robust, efficient, and reliable sentiment analysis. Comparing the hybrid technique to separate approaches, the outcomes demonstrate that the hybrid M-LSTM method considerably advances the performance of sentiment analysis. The suggested M-LSTM model's performance has also been contrasted with the latest state-of-the-art techniques for assessment metrics such as accuracy, F-1 measure, recall, precision, and area under the curve. Every one of the five evaluation parameters shows a significant improvement. The proposed model will be developed further in the following areas: (1) using sentiment classification results to summarize customer assessments; (2) evaluating the model's performance across numerous datasets; and (3) investigating additional NLP techniques to build a language-independent model.

REFERENCES

- [1] Alamanda MS. Aspect-based sentiment analysis search engine for social media data. *CSI Trans ICT*. 2020;8(2):193–7.
- [2] Alqaryouti O, Siyam N, Monem AA, Shaalan K. Aspect-based sentiment analysis using smart government review data. *Applied Computing and Informatics*. 2019. p 1–12.
- [3] Aslam N, Rustam F, Lee E, Washington PB, Ashraf I. Sentiment analysis and emotion detection on cryptocurrency related Tweets using ensemble LSTM-GRU Model. *IEEE Access*. 2022;10:39313–24.
- [4] Ayyub K, Iqbal S, Munir EU, Nisar MW, Abbasi M. Exploring diverse features for sentiment quantification using machine learning algorithms. *IEEE Access*. 2020;8:142819–31.
- [5] Bie Y, Yang Y. A multitask multiview neural network for end-to-end aspect-based sentiment analysis. *Big Data Mining Analytics*. 2021;4(3):195–207.

- [6] Chandra R, Kulkarni V. Semantic and sentiment analysis of selected Bhagavad Gita translations using BERT-based language framework. *IEEE Access*. 2022;10:21291–315.
- [7] Datta S, Chakrabarti S. Aspect based sentiment analysis for demonetization tweets by optimized recurrent neural network using fre fy-oriented multi-verse optimizer. *Sadhana*. 2021;46(2):1–23
- [8] Flyand. A hybrid framework for sentiment analysis using genetic algorithm based feature reduction. *IEEE Access*. 2019;7:14637–52.
- [9] Hao Y, Mu T, Hong R, Wang M, Liu X, Goulermas JY. Cross-domain sentiment encoding through stochastic word embedding. *IEEE Trans Knowl Data Eng*. 2020;32(10):1909–22
- [10] Hitkul S, Shahid S, Singhal D, Mahata P, Kumarguru RR, Shah. Aspect-Based Sentiment Analysis of Financial Headlines and Microblogs. In: *Deep Learning-Based Approaches for Sentiment Analysis*, Springer; 2020. pp 111–137
- [11] Huang F, Li X, Yuan C, Zhang S, Zhang J, Qiao S. Attention-emotion-enhanced convolutional LSTM for sentiment analysis. *IEEE Trans Neural Netw Learn Syst*. 2022;33(9):4332–45.
- [12] Khan A, Gul MA, Zareei M, Biswal RR, Zeb A, Naeem M, Saeed Y, Salim N. Movie review summarization using supervised learning and graph-based ranking algorithm. *Comput Intell Neurosci*. 2020;2020:1–14.
- [13] Khan J, Jeong BS. Summarizing customer review based on product feature and opinion. In: *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, 2016. p. 158–65.
- [14] Kumar R, Pannu HS, Malhi AK. Aspect-based sentiment analysis using deep networks and stochastic optimization. *Neural Comput Appl*. 2020;32(8):3221–35.
- [15] Li N, Chow CY, Zhang JD. SEML: a semi-supervised multi-task learning framework for aspect-based sentiment analysis. *IEEE Access*. 2020;8:189287–97.
- [16] Londhe A, Rao PVRDP. Aspect Based Sentiment Analysis – An Incremental Model Learning Approach Using LSTM-RNN. In: Singh, M., Tyagi, V., Gupta, P.K., Flusser, J., Ören, T., Sonawane, V.R. (eds) *Advances in Computing and Data Sciences. ICACDS 2021. Communications in Computer and Information Science*, vol 1440. 2021; Springer, Cham.
- [17] Lu Q, Zhu Z, Zhang G, Kang S, Liu P. Aspect-gated graph convolutional networks for aspect-based sentiment analysis. *Appl Intell*. 2021;51(7):4408–19.
- [18] Moussa ME, Mohamed EH, Haggag MH. A survey on opinion summarization techniques for social media. *Fut Comput Inform J*. 2018;3(1):82–109.
- [19] Muhammad PF, Kusumaningrum R, Wibowo A. Sentiment Analysis Using Word2vec and Long Short-Term Memory (LSTM) for Indonesian Hotel Reviews. *Proc Int Conf Computer Sci Computat Intell Procedia Computer Sci*. 2021;179:728–35.
- [20] Munuswamy S, Saranya MS, Ganapathy S, Muthurajkumar S, Kannan A. Sentiment analysis techniques for social mediabased recommendation systems. *Nat Acad Sci Lett*. 2021;44(3):281–7.
- [21] Oyeboode O, Alqahtani F, Orji R. Using machine learning and thematic analysis methods to evaluate mental health apps based on user reviews. *IEEE Access*. 2020;8:111141–58.
- [22] Prathi JK, Raparathi PK, Gopalachari MV. Real-time aspect-based sentiment analysis on consumer reviews. *Data Eng Commun Technol*. 2020;1079:801–10.
- [23] Saif H, Fernández M, He Y, Alani H. Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the STSGold. *ESSEM@AI*IA*. 2013.
- [24] Schouten K, Frasincar F. Survey on aspect-level sentiment analysis. *IEEE Trans Knowl Data Eng*. 2016;28(3):813–30.

- [25] Schouten K, van der Weijde O, Frasincar F, Dekker R. Supervised and unsupervised aspect category detection for sentiment analysis with co-occurrence data. *IEEE Trans Cybern.* 2018;48(4):1263–75
- [26] Shahana PH, Omman B. Evaluation of features on sentimental analysis. *Procedia Computer Sci.* 2015;46:1585–92.
- [27] Shams M, Khoshavi N, Baraani-Dastjerdi A. LISA: language-independent method for aspect-based sentiment analysis. *IEEE Access.* 2020;8:31034–44.
- [28] Singh M, Jakhar AK, Pandey S. Sentiment analysis on the impact of coronavirus in social life using the BERT model. *Soc Netw Anal Min.* 2021;11(1):11–33. H
- [29] Wang D. Coarse alignment of topic and sentiment: a unified model for cross-lingual sentiment classification. *IEEE Trans Neural Netw Learn Syst.* 2021;32:736–47.
- [30] Wang L, Niu J, Yu S. SentiDif: combining textual information and sentiment diffusion patterns for twitter sentiment analysis. *IEEE Trans Knowl Data Eng.* 2020;32(10):2026–39