Bitlis Eren Üniversitesi Fen Bilimleri Dergisi
BİTLİS EREN UNIVERSITY JOURNAL OF SCIENCE
ISSN: 2147-3129/e-ISSN: 2147-3188
VOLUME: 12 NO: 3 PAGE: 712-724 YEAR: 2023
DOI:10.17798/bitlisfen.1288561

Enhancing Deep Learning-Based Sentiment Analysis Using Static and Contextual Language Models

Khadija MOHAMAD[1], Kürşat Mustafa KARAOĞLAN[2*]

[1,2]*Karabük University, Faculty of Engineering, Department of Computer Engineering, 78050, Karabük, Türkiye*
(ORCID: 0009-0005-2741-3897) (ORCID: 0000-0001-9830-7622)

Keywords: Deep learning, Sentiment analysis, Static language models, Contextual language models, BERT, ELMo.

Abstract
Sentiment Analysis (SA) is an essential task of Natural Language Processing and is used in various fields such as marketing, brand reputation control, and social media monitoring. The various scores generated by users in product reviews are essential feedback sources for businesses to discover their products' positive or negative aspects. However, it takes work for businesses facing a large user population to accurately assess the consistency of the scores. Recently, automated methodologies based on Deep Learning (DL), which utilize static and especially pre-trained contextual language models, have shown successful performances in SA tasks. To address the issues mentioned above, this paper proposes Multi-layer Convolutional Neural Network-based SA approaches using Static Language Models (SLMs) such as Word2Vec and GloVe and Contextual Language Models (CLMs) such as ELMo and BERT that can evaluate product reviews with ratings. Focusing on improving model inputs by using sentence representations that can store richer features, this study applied SLMs and CLMs to the inputs of DL models and evaluated their impact on SA performance. To test the performance of the proposed approaches, experimental studies were conducted on the Amazon dataset, which is publicly available and considered a benchmark dataset by most researchers. According to the results of the experimental studies, the highest classification performance was obtained by applying the BERT CLM with 82% test and 84% training accuracy scores. The proposed approaches can be applied to various domains' SA tasks and provide insightful decision-making information.

1. Introduction

Sentiment Analysis (SA) identifies and extracts a text's underlying sentiment or opinion. As a result of the increase in textual data available on the internet, SA has become helpful in various applications, like market analysis, brand reputation management, social media monitoring, and news articles [1]. Deep Learning-based approaches have shown promising results in SA tasks [2], mainly using Static and Contextual Language Models. Static Language Models (SLMs), such as Word2Vec and GloVe, represent each word in a fixed-dimensional vector space based on its co-occurrence statistics. Contextual Language Models (CLMs), such as Embeddings from Language Models (ELMo) and Bidirectional Encoder Representations from Transformers (BERT), generate word representations that capture the context and meaning of the entire sentence [3].

SA studies, whose use has increased in recent years, are widely used in many areas, such as using SA in political analysis [4], [5], and social media monitoring [6]. In addition, studies to detect sentiment in social media are also presented in the literature [7], [8]. At the same time, in the marketing field, some studies detect sentiment in the reviews of

* Corresponding author: kkaraoglan@karabuk.edu.tr
Received: 27.04.2023, Accepted: 08.09.2023

customers or their feedback [9]. The role of SA also played a lot during the Corona pandemic period, as many researchers analyzed the feelings and emotions of people towards this disease during this period [10], [11].

Deep Learning has a significant role in Natural Language Processing (NLP) projects in multiple areas, such as entity recognition [12], [13], question answering [14], [15], and SA [16]. It has been observed that sentiment in the text can be detected using the Convolutional Neural Network (CNN) model [17], [18]. A Long Short-Term Memory (LSTM) [19], [20], on the other hand, is a neural network developed as a solution to the disappearing gradient problem that complicates the training of this data. In addition, there is research based on the Transformer Model to achieve this task. For example, emotions are classified using BERT as a Transformer-based Deep Learning model. It was found that it is possible to obtain a high classification accuracy of 88% [21], [22]. In addition, machine learning has a significant role in applying this task; for example, Support Vector Machine can be used to perform SA [23], [24]. In addition, some of the experiments resorted to linguistic analysis features, which showed beneficial results in the fields of text classification and context understanding [25].

As for the representation of words, research in this field has been numerous and varied to achieve this task. Some of them followed SLMs, and some of them followed CLMs; examples of SLMs are following FastText to represent words, which is the famous language model (word embedding) [26], and Word2Vec [27], but CLM refers to the process of representing words or phrases in a sentence based on their surrounding context, such as BERT [28], and Robustly Optimized BERT Pre-Training Approach (RoBERTa) [29], which uses Deep Learning techniques to generate CLMs by processing text in a way that takes into account the words that come before and after each word being analyzed. Multiple experiments that combined two methods of word embedding, such as combining GloVe and FastText, showed promising results from their combination [30].

Several studies have been conducted in the literature on SA. One study utilized [26] FastText for feature extraction and CNN as the classifier model to analyze sentiment in the Movie Reviews dataset, achieving exceptional accuracy. In another study [27], SA was performed using hotel reviews in Indonesia. In the given study, Word2Vec was used for feature extraction, LSTM was used as a classifier model, and high accuracy scores were obtained. Another study [20] used Residual Long Short-Term Memory and obtained acceptable accuracy scores for SA. In addition to the above studies, the Amazon dataset used in this study contains a wide variety of data and covers a wide range of diversity. Thus, this dataset is valuable for researchers to evaluate model performance. The literature [31] presents a hybrid approach combining SVM and k-Means to perform SA tasks on the Amazon dataset. Another study [32] focuses on cell phone reviews on Amazon dataset to perform SA tasks with the BERT language model. A similar study [33] applied a CNN-based SA approach using Word2Vec in text representation for SA on cell phone reviews. Other work in the literature [34], which uses TF-IDF for feature extraction and LSTM as a classifier model, also produces effective performance results on the Amazon dataset. These studies contribute to SA research in different areas by developing or using various feature extraction techniques and classifier models.

It is also worth noting that different studies have adopted different labeling approaches when working with the dataset in question. For example, in one study in the literature [32], comments with 1 and 2 stars were classified as negative, while comments with three stars were considered neutral, and comments with 4 and 5 stars were considered positive. In contrast, another study [35] turned the classification task into a binary problem by treating (1 and 2) stars as negative and (4 and 5) stars as positive, effectively excluding three-star reviews.

This simplified binary classification achieved higher accuracy than the more complex task of classifying the sentiment into five different categories, which is the approach adopted in our study. Moreover, several studies have extensively studied the SA of customer reviews from different sources. In one notable study [36], researchers collected customer reviews on “We Chat” for three years. These comments were associated with ratings ranging from 1 to 5 stars, where ratings of 1 and 2 were considered negative comments, 3 represented neutral feedback, and 4 and 5 indicated positive comments.

This paper addresses the growing need for robust, state-of-the-art SA models that accurately classify sentiments across various domains. The exponential growth of online reviews and customer feedback emphasizes the importance of understanding emotions expressed in textual data. However, obtaining accurate sentiment classification remains a significant challenge in this context. To address this challenge, in this study, we propose a Deep Learning-based SA approach that uses SLMs

and CLMs to effectively classify sentiments in textual data. This study examines the effectiveness of a multi-layer CNN architecture as a Deep Learning model for sentiment classification. The objective is to classify input data according to sentiments such as "very satisfied", "satisfied", "neutral", "unsatisfied", or "very unsatisfied". To evaluate the performance of our approaches, we conduct assessments using Amazon data, a publicly available SA dataset that includes a customer review dataset.

The main objectives of this paper are:

- Introducing a novel approach to SA: The study proposes new approaches using Deep Learning-based methodologies that utilize SLMs and CLMs. These approaches are evaluated on the Amazon dataset, which is publicly available for SA and is considered by most researchers as a benchmark dataset.
- Comparing the performance of static and contextual language models: This study compares the performance of SLMs such as Word2Vec and GloVe with CLMs such as ELMo and BERT for SA purposes.
- Development of a CNN for SA: The paper proposes CNN models that can use static or contextual representations of texts as input. The CNN models are trained to predict labels based on user-generated ratings in review texts.
- Obtaining high accuracy for SA: Among the proposed approaches, the BERT CLM-based approach achieves high classification results with 82% test accuracy and 84% training accuracy.
- Providing insights for decision-making: The proposed approach provides insightful decision-making information for various areas where SA is required, such as marketing, brand reputation control, and social media monitoring. And for businesses facing high user populations, it allows them to use user reviews and satisfaction scores to discover the positives or shortcomings of their products.

This paper is structured in 6 sections to present the research. Section 1 introduces the objectives of the study and lays the foundation for the following sections by providing a comprehensive literature review on neural network-based word representation models and SA analysis. Section 2 summarizes the specific research objectives by highlighting the pool of problems addressed in the study. Section 3 details the methodologies for this work and describes the specific approaches and techniques applied and developed to perform the experiments. Section 4 presents information about the dataset used in the study, the performance metrics applied, and the hyperparameters identified during the CNN model training process. Furthermore, the results obtained from the experiments are presented in detail in Section 4. In Section 5, a comprehensive description of the findings of the experimental studies is presented along with analysis and discussion. In the same chapter, key insights and observations from the experimental results are also presented. Finally, in Section 6, discussions and conclusions are given. In this section, the main findings of the study are summarized, and the performance results are discussed. Furthermore, this section concludes the paper by summarizing potential work for future research and advancements in the field of SA and word representation.

2. Research Objectives

This research aims to achieve the following objectives:

- Performing precise text pre-processing with NLP techniques without distorting the meaning and structure of the input texts
- Performing labeling operations to prepare model labels based on the customer score, considering the semantics of the review texts
- Generating vectors of review texts with contextual features using SLMs and pre-trained CLMs and ensuring that the inputs to the deep network model are transformed into the appropriate form
- Developing a CNN model for sentiment classification, and optimizing the fine-tuning of hyperparameters to improve the performance of the models
- Conducting experimental studies to measure the performance of the realized language models and evaluating the results of experimental studies.

To achieve the research goals outlined in this chapter, we conducted experiments using state-of-the-art contextual word representation models (CLMs) as input and CNN-based approaches. The experiments aim to compare static representation models with contextual models and test their performance evaluations. Through these experiments, we demonstrate the effectiveness of contextual word representation models over static models in category-based multi-class in SA tasks.

3. Methodology

In this section, the main steps we follow to create our SA system are explained, including language models for the representation of textual reviews and the Deep Learning model used in customer review

classification. Section 3.1 presents the data pre-processing, Section 3.2 gives the pre-trained models as language models used in this study, and Section 3.3 provides an overview of the architecture of the applied learning model.

3.1. Pre-processing

Applying measures like text cleaning or preprocessing is crucial for enhancing the performance of our approaches, and it constitutes an essential step in any NLP project. Removing ineffective words or punctuation marks must be executed precisely, ensuring the preservation of sentence structure, integrity, or opinion while aligning with the intended scenario [37].

Punctuation, URLs (https:// or www.), numbers, and symbols (#tags, mentions, and emojis) were all removed from the reviews as they do not convey sentiment. Subsequently, the reviews underwent processes such as lowercasing, lemmatization, and tokenization. Tokenization was performed using the tokenizer from the NLTK package. Finally, stop words were dropped from the text using the NLTK package to obtain the refined list.

3.2. Static and Contextual Language Models

Word representations are numerical representations of words that capture their semantic and syntactic information. These embeddings are used as inputs to Deep Learning-based SA models. This paper uses two types of embeddings: static embeddings with Word2Vec and GloVe as SLMs and contextual embeddings with BERT and ELMo as CLMs.

SLMs are pre-trained on a large corpus of text and represent each word in a fixed-dimensional vector space based on its co-occurrence statistics. These models detect the distributive properties of words and are useful for capturing semantics at the word level. There are several SLMs, such as Word2Vec and GloVe. Word2Vec is a neural-based model that learns embedding vectors by predicting the surrounding words in a context [38]. GloVe, on the other hand, is a count-based model that uses the co-occurrence matrix of words to generate embeddings [39]. We choose 100 as the vector dimension to represent each word in the case of SLMs.

CLMs are generated by Deep Learning models that consider the context and meaning of the entire sentence. These models are beneficial for detecting the nuances of language, such as irony and negation. Two popular models for generating contextual embeddings are ELMo and BERT. ELMo uses a bi-directional LSTM to generate an embedding vector that captures the context of the sentence [40]. BERT uses a transformer-based architecture to generate embeddings that capture the sentence's syntax and semantics [41]. From BERT models, we chose the $BERT_{large\text{-}uncased}$ model (BERTM) for creating vectors. The vector dimension we have in the case of CLMs is 1024 to represent each review in our dataset.

3.3. Multi-layer Convolutional Neural Network

CNN, one of the Deep Neural Network models, is usually used in computer vision and image recognition tasks, but it has also been successfully applied in SA and other NLP tasks. The network architecture of a CNN comprises several layers, such as convolutional, pooling, and fully connected layers. The input data for a CNN is typically a two-dimensional matrix. The convolutional layers apply a series of learnable filters to the input data to produce feature maps that indicate the existence of particular patterns and features in the data. The feature maps are then down-sampled by the pooling layers to make them smaller while preserving the most crucial data. The fully connected layers produce the network's ultimate output after they have processed the pooling layers' flattened output.

A CNN can capture the local context and relationships between words in a sentence in the context of SA. Convolutional layers can be used to create feature maps that show the presence of particular word or phrase combinations in the input text by applying them to the pre-trained word embedding vectors of the input words. The fully connected layers can then process the pooling layers' flattened output to provide the input text's final sentiment prediction. At the same time, the pooling layers can down-sample the feature maps to collect the most crucial information. CNNs are potent tools for SA tasks because they are good at catching local and global correlations between words in the text.

In this study, a multi-layer CNN model was created and trained for various epochs according to the language model and cross-validated using k-fold 5. Since the task involves multiple classifications, the loss function was set to categorical cross-entropy. The CNN model architecture and hyper-parameters used in the study are presented in Table 2. These values were obtained by optimizing through experience to achieve acceptable accuracy while at the same time taking into account overfitting and underfitting.

In order to take into account the multi-label classification included in this study, the SoftMax

function was chosen as the output function. The formulas applied to the ReLU and SoftMax functions are presented in equations (1) and (2).

$$ReLU(x) = \max(0, x) \quad (1)$$

$$SoftMax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \quad (2)$$

Where x is the input value, n is the number of classes, and zi is the value of the input vector to the SoftMax function.

3.4. Architecture of the Proposed Approaches

In constructing the architecture of the proposed approaches, the initial step involved aligning the input data to the model inputs, and ensuring data quality and consistency through data preprocessing techniques applied to the reviews. Semantic vectors were then constructed using SLMs and CLMs to extract rich semantic information from the processed data, allowing the underlying meaning and context of the reviews to be effectively captured. A CNN-based classification model was employed based on semantic vectors, chosen for its capability to leverage the hierarchical representation of features in semantic vectors.

To mitigate the risks of overfitting and underfitting, a rigorous k-fold cross-validation methodology was utilized to optimize the model's performance, with k set to five, enabling iterative training and validation of the model using different subsets of the training data. By employing k-fold cross-validation, the model's generalization ability was enhanced, allowing it to perform well on unseen data. Throughout the iterative training and validation process, consideration was given to addressing overfitting and underfitting issues. Overfitting ensues when the model performs exceptionally well on the training data but fails to generalize to new, untouched data. In contrast to overfitting, underfitting ensues when the model's sophistication is insufficient to capture the underlying patterns in the data, resulting in poor performance.

In addition to the above overfitting and underfitting checks, the K-fold cross-validation process analyzed the model's performance on both the training and validation sets and fine-tuned the hyperparameters to ensure the optimal configuration of the model. After the training phase, the performance of the trained model was evaluated on the test data, and fundamental and valid performance metrics measured the outputs. These metrics, such as accuracy, precision, recall, and F_1-score, produced quantitative analyses of the model's effectiveness in capturing desired patterns and predicting the precision of reviews or other relevant aspects. Figure 1 shows the architecture for implementing the proposed approaches and their detailed components.

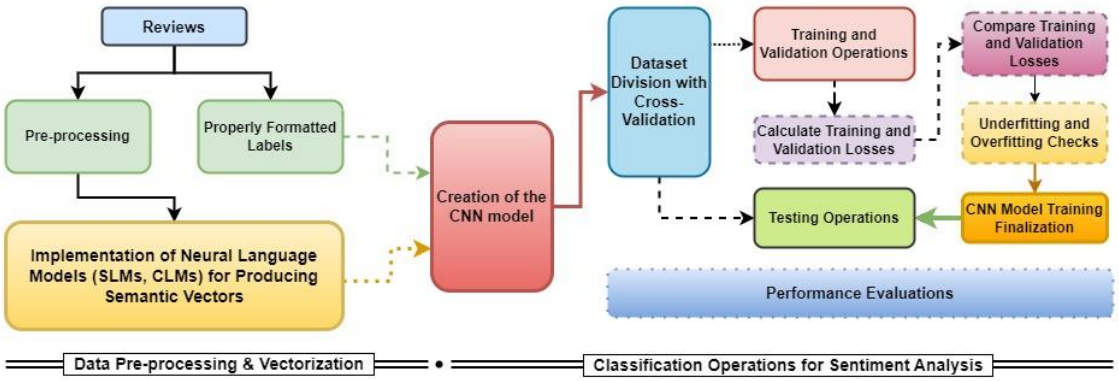

Figure 1. The architecture with components for the execution of the proposed approaches

4. Experimental Study

This section summarizes the experimental studies, covering the dataset and training hyperparameters. The results are then presented using various performance metrics. Specifically, Section 4.1 provides detailed information about the dataset employed, classifier hyperparameters, and performance metrics utilized in the experimental settings. Afterward, Section 4.2 offers a comprehensive analysis of the experimental results, including performance comparisons.

4.1. Experimental Settings

4.1.1. Dataset

In this subsection, information about the data set used in the experimental studies is described in

detail. The experiments were conducted using Amazon datasets comprising mobile phone reviews totaling 30,000 customer reviews, each accompanied by a corresponding rating value. Prior to the initiation of the training process, the dataset was divided into Train and Test sets. Notably, cross-validation was applied to parse the data, a technique utilized to ensure the robustness and reliability of our results.

Table 1 presents descriptive statistical information about the dataset utilized in our study. For example, the table indicates that there are 16,251 reviews labeled as "very satisfied," as these reviews received a rating of 5. Our classification scheme categorizes 1-star reviews as "very unsatisfied," 2-star reviews as "unsatisfied," 3-star reviews as "neutral," 4-star reviews as "satisfied," and 5-star reviews as "very satisfied". The table provides a comprehensive overview of the sentiment label distribution based on the corresponding customer ratings.

Table 1. Statistical descriptive information about the dataset used

Total number of reviews	30000				
Shortest review	1 word				
Longest review	250 words				
Average word count	30				
Number of labels	5				
Distribution of each class	very satisfied	satisfied	neutral	unsatisfied	very unsatisfied
	16251	4300	2290	1910	5249

4.1.2. Evaluation Metrics

Various metrics used to evaluate multiple classifications are presented in the literature, including micro, macro, and example-based average metrics. Micro metrics [42] have been widely adopted in the literature to assess multiple classifications on large-sized data, showcasing their effectiveness in handling multiple classes. Additionally, alternative metrics can be employed to evaluate different systems.

In this study, a comprehensive set of metrics was employed to assess the performance of our Deep Learning model, encompassing Accuracy (3), Recall (4), Precision (5), and F_1-score (6), Learning curves, and Receiver Operating Characteristic (ROC). These metrics allow for a holistic evaluation of the model's effectiveness in capturing relevant patterns and predicting sentiment sensitivity in the reviews.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$F1 - score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \quad (6)$$

Whereas "TP" denotes the number of True Positives, "TN" denotes the number of True Negatives, "FP" denotes the number of False Positives, and "FN" denotes the number of False Negatives.

4.1.3. Training Hyperparameters of Deep Learning Approaches

The user-specified parameters employed to train the Deep Learning model are referred to as hyperparameters. Hyperparameters govern the model's behavior and significantly affect its performance. Configuring these values is crucial for achieving optimal results in the model training process [43].

In our study, the hyperparameter settings were carefully fine-tuned, and the selected values are presented in Table 2. The table presents a comprehensive overview of the determined hyperparameters, facilitating the fine-tuning of the model for enhanced performance and effective learning during training. These hyperparameter settings are essential in ensuring that the model's behavior aligns with the specific requirements of the task, producing reliable and accurate results.

Table 2 presents the hyperparameters utilized in our study, encompassing various essential aspects of the model configuration. The hyperparameters include the number of epochs, batch size, loss function, activation function, learning rate, CNN activation function, filter size,

kernel size, dropout rate, training approach, and optimizer. The number of epochs represents the frequency with which the training dataset is used during the training process. Batch size, however, denotes the number of samples fed into the model at once for each iteration.

The loss function measures the discrepancy between predicted and actual sentiment labels. The activation function generates a probability distribution over the output classes, enabling the model to make effective predictions.

Table 2. Hyperparameters settings for the applied CNN model

Hyperparameters	**Properties**			
Number of epochs	BERTM	ELMo	Word2Vec	GloVe
	48	53	29	25
Batch size	256			
Loss function	Categorical Cross-entropy			
Activation functions	Softmax			
Learning rate	0.0001			
Activation function of CNNs	RelU			
Filter size of CNN_1	50			
Filter size of CNN_2	100			
Filter size of CNN_3	200			
Kernel size of CNNs	3			
Dropout of CNNs	0.2			
Train Approach	Cross-validation			
Optimizer	Adam			

Meanwhile, the learning rate governs the step size during the optimization process, impacting the convergence and stability of the model. Additionally, the CNN activation function applies explicitly to the convolutional layer, enhancing the feature extraction capability of the model.

The CNN filter size determines the number of filters employed in the convolutional layer, while the kernel size indicates the dimensions of the kernel used in the convolutional layer. A dropout technique is employed to prevent overfitting, reducing the likelihood of the model relying too heavily on specific features during training. Furthermore, our training approach involves cross-validation, ensuring our results' reliability and generalization. As for optimization, the Adam optimizer is employed, aiding in the efficient convergence of the model during the training process.

4.2. Experimental Results with Performance Comparison

The learning curves serve as essential diagnostic tools for evaluating the model's convergence and identifying potential issues such as overfitting or underfitting. They provide a comprehensive understanding of the model's behavior during training, enabling researchers to fine-tune the hyperparameters and optimize the model's performance.

Figure 2 displays the learning curves of the training and validation phases, utilizing BERTM as the word representation model. These learning curves provide valuable insights into the model's performance over time, allowing a comprehensive assessment of its convergence and generalization capabilities. Similarly, Figure 3 visually represents the learning curves for training and validation, showing the model's behavior when employing ELMo as the word representation model. Likewise, Figure 4 showcases the learning curves for training and validation using Word2Vec, presenting a detailed advancement of the model's performance throughout the training process. Lastly, Figure 5 displays the learning curves for GloVe as the word representation model, facilitating a thorough comparison of its performance during the training and validation phases.

Table 3 presents the performance results of SA classification approaches developed using SLMs and CLMs according to various metrics. The values indicate the performance quality of each model according to the experimental studies performed.

Table 3. Performance metrics results of approaches developed according to SLMs and CLMs

Metrics	BERTM	ELMo	Word2Vec	GloVe
Accuracy	0.82	0.80	0.75	0.78
Precision	0.82	0.79	0.72	0.76
Recall	0.81	0.80	0.75	0.78
F_1-score	0.81	0.79	0.73	0.77

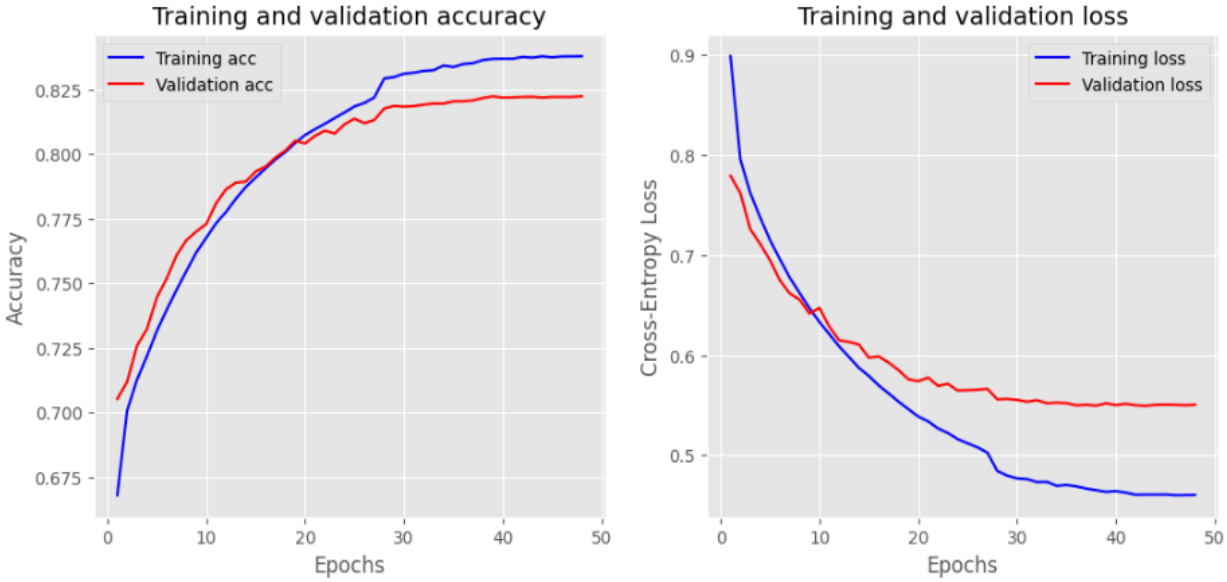

Figure 2. Learning curves of training and validation of SA with the BERTM

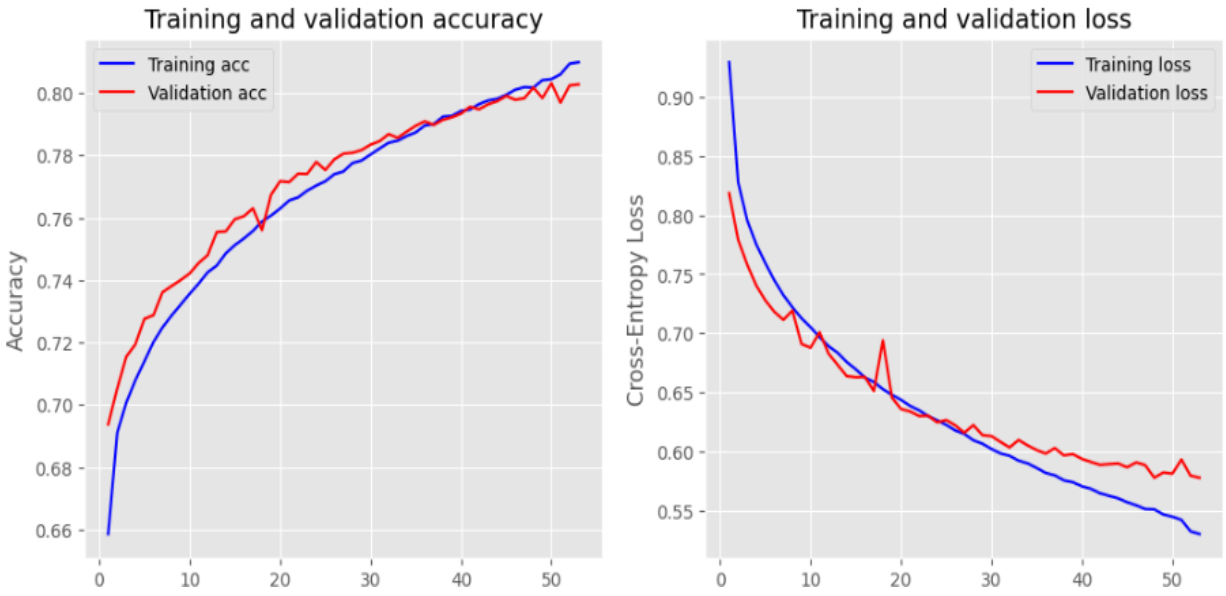

Figure 3. Learning curves of training and validation of SA with the ELMo model

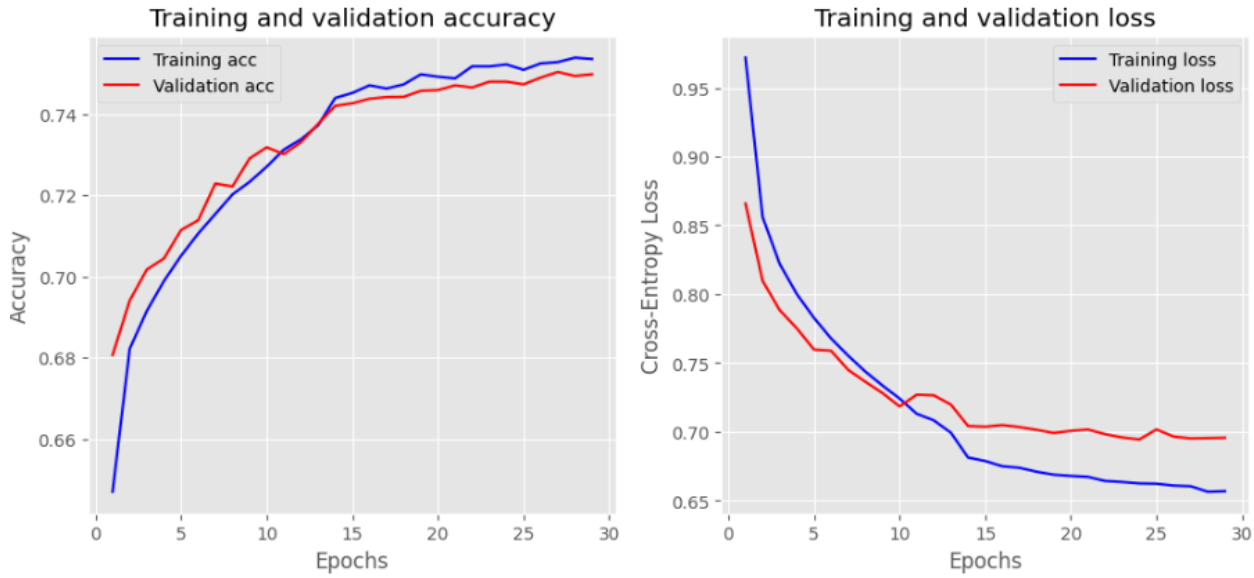

Figure 4. Learning curves of training and validation of SA with the Word2Vec model

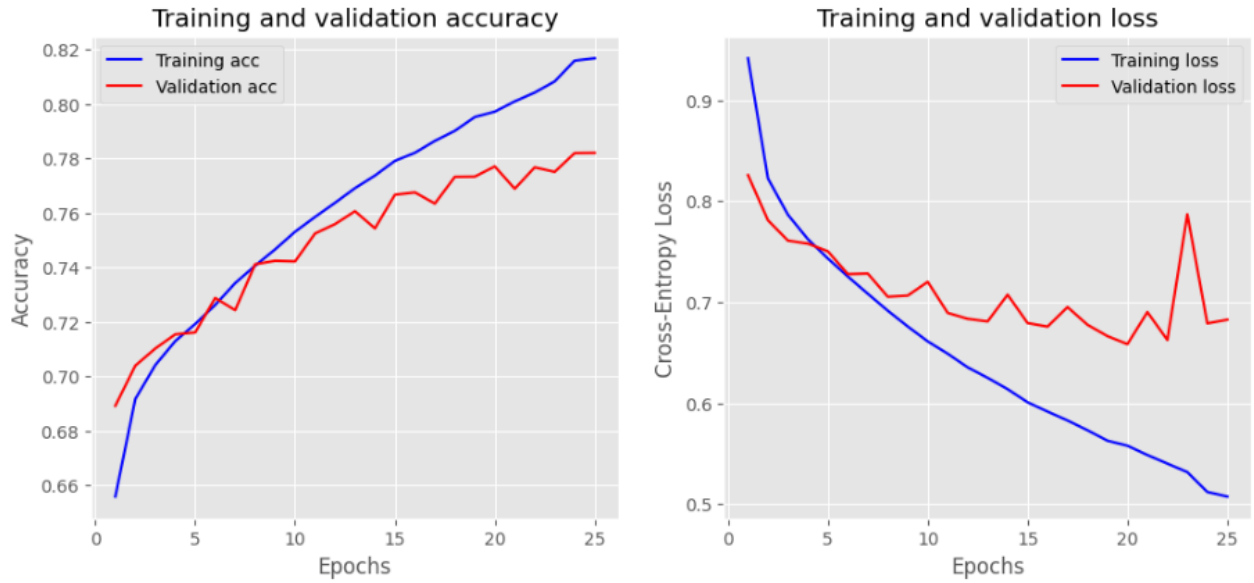

Figure 5. Learning curves of training and validation of SA with the GloVe model

Figure 6 illustrates the ROC curve plots obtained for each neural-based language model, presenting a more precise and comprehensive evaluation of the performance quality in the experimental studies. The ROC curves showcase the model's ability to discriminate between different sentiment classes, allowing for a visual comparison of their respective classification performances.

5. Discussion and Results

This section presents our understanding of the performance of both SLMs and CLMs. It describes the challenges the proposed SA approaches face in attempting to classify more than just positive or negative categories.

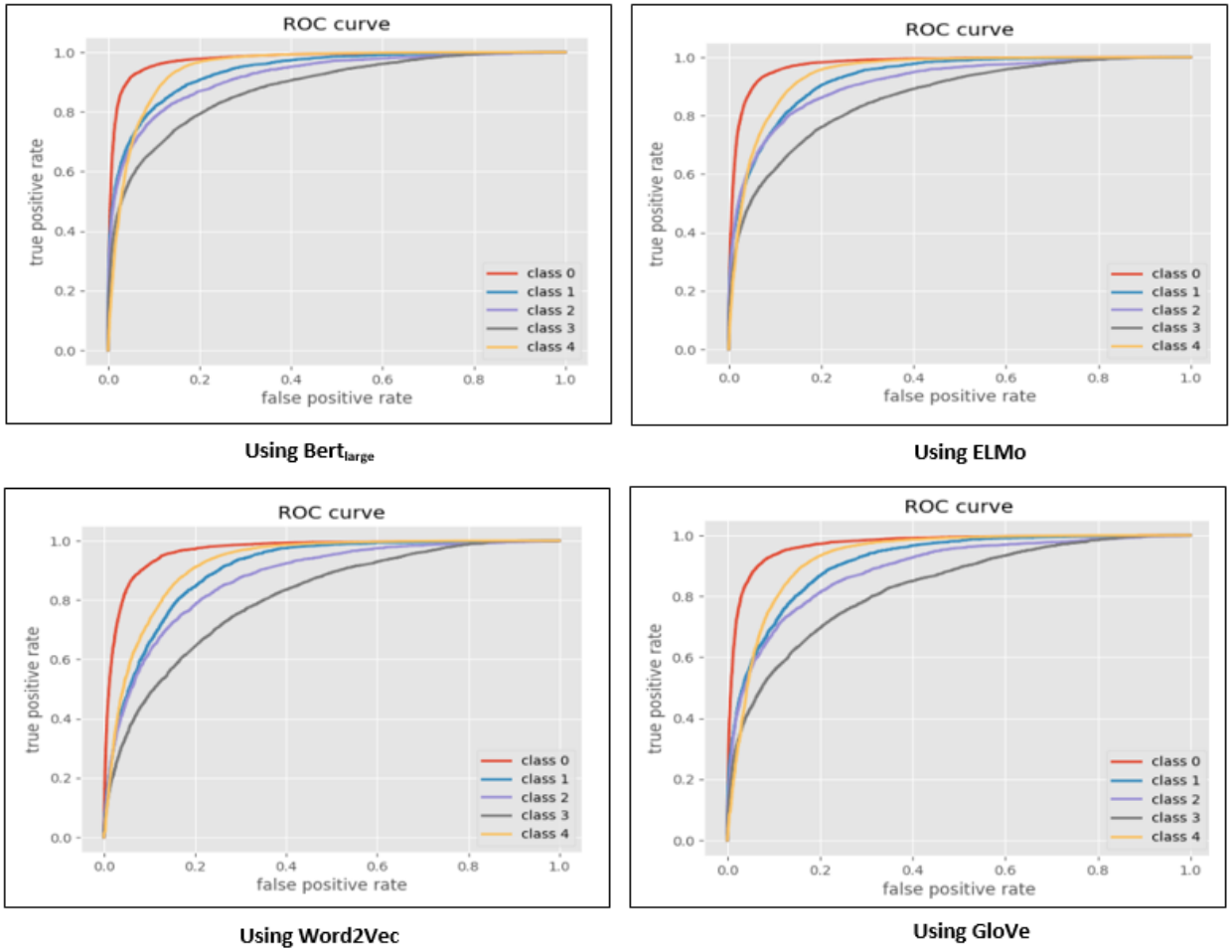

Figure 6. ROC curve plots showing the classification performance of the applied language models.

Based on the ROC curves, it is evident that categorizing classes with ratings of 1 (unsatisfied), 2 (neutral), and 3 (satisfied) poses a significant challenge. Their classification accuracy is lower compared to categories with ratings of 0 (very unsatisfied) and 4 (very satisfied). On the other hand, classes belonging to categories 0 and 4 are easily classified with higher accuracy.

The study reveals that BERTM outperforms other commonly utilized language models with remarkable accuracy and recall values of 0.82. Additionally, when ELMo is employed as the language model, elevated levels of accuracy are achieved. Fundamentally, higher accuracy and performance are attained by utilizing CLMs instead of SLMs. However, classifying feelings into multiple classes, as in our case with five classes, presents challenges due to the closeness in meaning and confusion between emotions. For instance, classes classified as 2 and 3 exhibit similarities in feelings and opinions, making their differentiation difficult. The same issue applies between classes with a classification of 1 and 2, as evident from all the ROC curves in Figure 6. Regarding hyperparameters, several settings were experimented with during the training of the CNN model before obtaining the result. Instances of decreased accuracy were observed when the number of training epochs was set to 10.

Selecting the appropriate number of epochs is a crucial and meticulous task in training Deep Learning models. In our proposed model, we initially selected an epoch of 60, and then, for each language model, we carefully determined the suitable epoch number by observing the learning rate curves. Ultimately, the last suitable epoch number was determined through experimentation, as presented in Table 3. Through our experimentation, we observed that using GloVe as the language model, an overfitting problem occurs when the number of epochs exceeds 25. Similarly, when using Word2Vec, an overfitting problem arises when the number of epochs exceeds 29. For BERTM, the critical threshold is 48 epochs, beyond which overfitting becomes a concern. Lastly, when

utilizing ELMo, an overfitting problem occurs when the number of epochs exceeds 53.

Based on the findings of the experimental study, it has been observed that neural network-based word representation models, such as SLMs, exhibit limited precision in considering the specific context of words. In contrast, contemporary pre-trained approaches like CLMs demonstrate the ability to generate distinctive vector representations that can encompass a greater number of features by incorporating the context of a word within its originating sentence. Consequently, CLMs have the capacity to capture the intricate nuances of a word's meaning within its particular context. This implies that the rich vectors produced by CLMs outperform those of SLMs.

6. Conclusion

This paper proposes high-performance SA approaches in which Deep Learning and pre-trained sentence representation models are applied. In these approaches, four state-of-the-art pre-trained models are used to represent the input text data, including BERTM, ELMo as CLM, and Word2Vec and GloVe as SLM.

In conclusion, it was demonstrated that the semantic meaning and context of words in the text can be effectively captured, and the performance of SA systems can be improved by utilizing Deep Learning architectures in conjunction with language models. The superiority of modern language models like BERTM and ELMo over traditional word embeddings as SLMs was established in this study, as evidenced by their higher accuracy and superior performance in the task. Furthermore, the investigation highlighted the significance of hyperparameters in influencing model accuracy. Overall, the limitations of SA across multiple categories were underscored, emphasizing the advantages of employing CLMs that consider the specific context in which a word is employed. Furthermore, this study has shown that the performance of language models with different architectures and training sets significantly affects the classification operations when contextualizing input texts and vectorizing them according to a semantic and distributed space. It is also essential that the features stored in the semantic vectors transformed from texts are not lost and that their valuable aspects are strengthened and included in the learning algorithms. In this context, the performance of the CNN model as a classifier is evaluated in this study. Although the classification model proposed in this study offers high performance, it is thought that with Deep Learning models with attention mechanisms and advanced CNN architectures to be developed by considering fine-tuning optimizations, high applicability, and higher performance results can be obtained.

In the future, we plan to use our experience in this study to test the classification performance of the attention-learning models on state-of-the-art pre-trained language models, as well as to increase the resolution of feature extraction by combining the contextual vectors obtained from language models. In addition to the above, we also plan to classify the semantic vectors in SA by vectorizing them with different data types (e.g. audio, video, and text).

Contributions of the authors

This study benefited from the diverse expertise of its authors.

Karaoğlan's contributions encompassed conceptualization, research design, editing, supervision, project management, critical review, and final approval. Mohamad's role focused on literature review, data collection, data analysis, and manuscript composition.

Conflict of Interest Statement

There is no conflict of interest between the authors.

Statement of Research and Publication Ethics

The study is complied with research and publication ethics.

References

[1] J. Hartmann, M. Heitmann, C. Siebert, and C. Schamp, “More than a feeling: Accuracy and application c sentiment analysis”, *Int. J. Res. Mark.*, vol. 40, no. 1, pp. 75–87, 2023.

[2] H. T. Phan, N. T. Nguyen, and D. Hwang, “Aspect-level sentiment analysis: A survey of grap convolutional network methods”, *Inf. Fusion*, vol. 91, pp. 149–172, 2023.

[3] F. Lin, S. Liu, C. Zhang, J. Fan, and Z. Wu, "StyleBERT: Text-audio sentiment analysis with Bi-direction Style Enhancement", *Inf. Syst.*, vol. 114, no. 102147, p. 102147, 2023.

[4] M. M. Hasan and H. Jiang, "Political sentiment and corporate social responsibility", *Br. Account. Rev.*, vo 55, no. 1, p. 101170, 2023.

[5] D. Antypas, A. Preece, and J. Camacho-Collados, "Negativity spreads faster: A large-scale multilingu Twitter analysis on the role of sentiment in political communication", *arXiv [cs.CL]*, 2022.

[6] A. R. Rahmanti *et al.*, "Social media sentiment analysis to monitor the performance of vaccination coverag during the early phase of the national COVID-19 vaccine rollout", *Comput. Methods Programs Biomed* vol. 221, no. 106838, p. 106838, 2022.

[7] R. Haque, N. Islam, M. Tasneem, and A. K. Das, "Multi-class sentiment classification on Bengali soci media comments using machine learning", *International Journal of Cognitive Computing in Engineerin* vol. 4, pp. 21–35, 2023.

[8] C. Qian, N. Mathur, N. H. Zakaria, R. Arora, V. Gupta, and M. Ali, "Understanding public opinions o social media for financial sentiment analysis using AI-based techniques", *Inf. Process. Manag.*, vol. 5 no. 6, p. 103098, 2022.

[9] H.-C. K. Lin, T.-H. Wang, G.-C. Lin, S.-C. Cheng, H.-R. Chen, and Y.-M. Huang, 'Applying sentime analysis to automatically classify consumer comments concerning marketing 4Cs aspects', *Appl. So Comput.*, vol. 97, no. 106755, p. 106755, 2020.

[10] D. Sunitha, R. K. Patra, N. V. Babu, A. Suresh, and S. C. Gupta, "Twitter sentiment analysis using ensembl based deep learning model towards COVID-19 in India and European countries", *Pattern Recognit. Let* vol. 158, pp. 164–170, 2022.

[11] N. Leelawat *et al.*, "Twitter data sentiment analysis of tourism in Thailand during the COVID-19 pandem using machine learning", *Heliyon*, vol. 8, no. 10, p. e10894, 2022.

[12] M. Bhattacharya, S. Bhat, S. Tripathy, A. Bansal, and M. Choudhary, "Improving biomedical named entit recognition through transfer learning and asymmetric tri-training", *Procedia Comput. Sci.*, vol. 218, p 2723–2733, 2023.

[13] A. Goyal, V. Gupta, and M. Kumar, "A deep learning-based bilingual Hindi and Punjabi named entit recognition system using enhanced word embeddings", *Knowl. Based Syst.*, vol. 234, no. 107601, 107601, 2021.

[14] Q. Qiu, M. Tian, K. Ma, Y. J. Tan, L. Tao, and Z. Xie, "A question answering system based on miner exploration ontology generation: A deep learning methodology", *Ore Geol. Rev.*, vol. 153, no. 105294, 105294, 2023.

[15] A. Al-Sadi, M. Al-Ayyoub, Y. Jararweh, and F. Costen, "Visual question answering in the medical domai based on deep learning approaches: A comprehensive study", *Pattern Recognit. Lett.*, vol. 150, pp. 57–7 2021.

[16] N. Sharm, T. Jain, S. S. Narayan, and A. C. Kandakar, "Sentiment analysis of Amazon smartphone review using machine learning & deep learning", in *2022 IEEE International Conference on Data Science an Information System (ICDSIS)*, 2022.

[17] D. Maity, S. Kanakaraddi, and S. Giraddi, "Text sentiment analysis based on multichannel convolution neural networks and syntactic structure", *Procedia Comput. Sci.*, vol. 218, pp. 220–226, 2023.

[18] W. Li, L. Zhu, Y. Shi, K. Guo, and E. Cambria, "User reviews: Sentiment analysis using lexicon integrate two-channel CNN–LSTM family models", *Appl. Soft Comput.*, vol. 94, no. 106435, p. 106435, 2020.

[19] Y. Zhang, J. Wang, and X. Zhang, "Conciseness is better: Recurrent attention LSTM model for documen level sentiment analysis", *Neurocomputing*, vol. 462, pp. 101–112, 2021.

[20] D. O. Oyewola, L. A. Oladimeji, S. O. Julius, L. B. Kachalla, and E. G. Dada, "Optimizing sentime analysis of Nigerian 2023 presidential election using two-stage residual long short term memory", *Heliyo* vol. 9, no. 4, p. e14836, 2023.

[21] A. Patel, P. Oza, and S. Agrawal, "Sentiment analysis of customer feedback and reviews for airline service using language representation model", *Procedia Comput. Sci.*, vol. 218, pp. 2459–2467, 2023.

[22] M. P. Geetha and D. Karthika Renuka, "Improving the performance of aspect based sentiment analys using fine-tuned Bert Base Uncased model", *International Journal of Intelligent Networks*, vol. 2, pp. 64 69, 2021.

[23] A. Borg and M. Boldt, "Using VADER sentiment and SVM for predicting customer respons sentiment", *Expert Syst. Appl.*, vol. 162, no. 113746, p. 113746, 2020.

[24] T. H. Jaya Hidayat, Y. Ruldeviyani, A. R. Aditama, G. R. Madya, A. W. Nugraha, and M. W. Adisaputr "Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM an logistic regression as classifier", *Procedia Comput. Sci.*, vol. 197, pp. 660–667, 2022.

[25] M. Bibi *et al.*, "A novel unsupervised ensemble framework using concept-based linguistic methods an machine learning for twitter sentiment analysis", *Pattern Recognit. Lett.*, vol. 158, pp. 80–86, 2022.

[26] I. N. Khasanah, "Sentiment classification using fastText embedding and deep learning model", *Procedi Comput. Sci.*, vol. 189, pp. 343–350, 2021.

[27] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment analysis using Word2vec and lon short-term memory (LSTM) for Indonesian hotel reviews", *Procedia Comput. Sci.*, vol. 179, pp. 728–73 2021.

[28] K. Kaur and P. Kaur, "BERT-CNN: Improving BERT for requirements classification usin CNN", *Procedia Comput. Sci.*, vol. 218, pp. 2604–2611, 2023.

[29] M. Siddharth and R. Aarthi, "Blended multi-class text to image synthesis GANs with RoBerTa and Mas R-CNN", *Procedia Comput. Sci.*, vol. 218, pp. 845–857, 2023.

[30] N. Badri, F. Kboubi, and A. H. Chaibi, "Combining FastText and glove word embedding for offensive an hate speech text detection", *Procedia Comput. Sci.*, vol. 207, pp. 769–778, 2022.

[31] K. Korovkinas, P. Danėnas, and G. Garšva, "SVM and k-means hybrid method for textual data sentimer analysis", *Balt. J. Mod. Comput.*, vol. 7, no. 1, 2019.

[32] A. S. M. AlQahtani, "Product Sentiment Analysis for Amazon Reviews", *Int. J. Comput. Sci. Inf. Technol* vol. 13, no. 3, pp. 15–30, 2021.

[33] S. A. Aljuhani and N. Saleh, "A comparison of sentiment analysis methods on Amazon reviews of mobil phones", *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 6, 2019.

[34] Sangeetha and Kumaran, 'Sentiment analysis of amazon user reviews using a hybrid approach', *Measu Sens.*, vol. 27, no. 100790, p. 100790, 2023.

[35] B. Bansal and S. Srivastava, "Sentiment classification of online consumer reviews using word vecto representations", *Procedia Comput. Sci.*, vol. 132, pp. 1147–1153, 2018.

[36] L. Zhang, K. Hua, H. Wang, G. Qian, and L. Zhang, "Sentiment analysis on reviews of mobil users", *Procedia Comput. Sci.*, vol. 34, pp. 458–465, 2014.

[37] K. M. Karaoğlan and O. Fındık, "Extended rule-based opinion target extraction with a novel text pre processing method and ensemble learning", *Appl. Soft Comput.*, vol. 118, no. 108524, p. 108524, 2022.

[38] A. K. Sharma, S. Chaurasia, and D. K. Srivastava, "Sentimental short sentences classification by usin CNN deep learning model with fine tuned Word2Vec", *Procedia Comput. Sci.*, vol. 167, pp. 1139–114 2020.

[39] A. Pimpalkar and J. R. Raj R, "MBiLSTMGloVe: Embedding GloVe knowledge into the corpus usin multi-layer BiLSTM deep learning model for social media sentiment analysis", *Expert Syst. Appl.*, vo 203, no. 117581, p. 117581, 2022.

[40] M. Affi and C. Latiri, "BE-BLC: BERT-ELMO-based deep neural network architecture for English name entity recognition task", *Procedia Comput. Sci.*, vol. 192, pp. 168–181, 2021.

[41] A. Zhao and Y. Yu, "Knowledge-enabled BERT for aspect-based sentiment analysis", *Knowl. Based Syst* vol. 227, no. 107220, p. 107220, 2021.

[42] F. Gargiulo, S. Silvestri, M. Ciampi, and G. De Pietro, "Deep neural network for hierarchical extrem multi-label text classification", *Appl. Soft Comput.*, vol. 79, pp. 125–138, 2019.

[43] Z. A. Sejuti and M. S. Islam, "A hybrid CNN-KNN approach for identification of COVID-19 with 5-fol cross validation", *Sens. Int.*, vol. 4, no. 100229, p. 100229, 2023.