Iğdır Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 15(1), 39-56, 2025
Journal of the Institute of Science and Technology, 15(1), 39-56, 2025
ISSN: 2146-0574, eISSN: 2536-4618
Bilgisayar Mühendisliği / *Computer Engineering* **DOI: 10.21597/jist.1523220**
Araştırma Makalesi / *Research Article*
Geliş tarihi / *Received*: 27.07.2024 **Kabul tarihi / *Accepted*: 03.01.2025**

Atıf İçin: Çetiner, H. ve Metlek, S. (2025). SAHRAN: Dikkat Tabanlı Tekrarlayan Sinir Ağı ile Hotel Yorumlarının Duygu Analizi. *Iğdır Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 15(1), 39-56.

To Cite: Çetiner, H. & Metlek, S. (2025). SAHRAN: Sentiment Analysis of Hotel Reviews with Attention-Based Recurrent Neural Network. *Journal of the Institute of Science and Technology*, 15(1), 39-56.

SAHRAN: Dikkat Tabanlı Yinelemeli Sinir Ağı ile Hotel Yorumlarının Duygu Analizi

Halit ÇETİNER[1*], Sedat METLEK[2]

Öne Çıkanlar:
- BiGRU ve BiLSTM ağları
- Nokta çarpım tabanlı bir dikkat mekanizması
- Word2vec kelime gömme katmanı

Anahtar Kelimeler:
- RNN
- Derin öğrenme
- BiGRU
- BiLSTM
- Doğal dil işleme

ÖZET:

Bir kullanıcının herhangi bir amaç içerisinde bulunan web sayfasında ifade edeceği yorumları otomatik olarak duygu yönünden analiz etmek hızla genişleyen önemli bir araştırma alanıdır. Literatürdeki adıyla metin duygu analizi, herhangi bir amaç ile tanımlanan yorumlardaki kullanıcıların duygusal eğilimlerini belirleyebilmeyi sağlayan bir tekniktir. Tatil siteleri, alışveriş sayfaları, sosyal medya, marka yorumları, finans yorumları, sağlık siteleri, siyaset sayfaları gibi binlerce insanın faydalandığı web sayfalarındaki içeriklerin kullanıcılar tarafından yorumlanması gerçekleştirilmektedir. Yapılan yorumlar, herhangi bir şekilde bu hizmetlerden faydalanmak isteyen bir kullanıcıyı doğrudan etkileme özelliğine sahiptir. Bu sebeplerden dolayı yorumların otomatik incelenmesinde insanların yorumlarındaki duygularını incelemek önem arz etmektedir. Yinelemeli Sinir Ağı (RNN) tabanlı mimariler Doğal Dil İşleme (NLP) problemlerinin çözümünde dikkat çekici başarılar sağlamıştır. Bu makale kapsamında tripadvisor web sayfasından elde edilen halka açık bir veriseti üzerinde çalışıp duygu analizi gerçekleştiren RNN tabanlı bir derin öğrenme modeli önerilmiştir. Önerilen SAHRAN modeli, kullanıcı yorumlarındaki duygusal sözcükleri yakalayabilmek için nokta çarpım yapısını temel alan bir dikkat mekanizması kullanılmıştır. Modelde, duygu özelliklerini yakalayabilmek için de Çift Yönlü Kapılı Yinelemeli Hücreler (BiGRU) ve Çift Yönlü Uzun Kısa Süreli Bellek (BiLSTM) derin öğrenme katmaları modele entegre edilmiştir. Yapılan deneysel çalışmalar neticesinde önerilen SAHRAN modeli hassasiyet, geri çağırma, F1 puanı ve doğruluk performans ölçütleri açısından sırasıyla 0.9524, 0.9685, 0.9082 ve 0.9338 performans değerlerini elde etmiştir.

SAHRAN: Sentiment Analysis of Hotel Reviews with Attention-Based Recurrent Neural Network

Highlights:
- BiGRU and BiLSTM networks
- A dot product based attention mechanism
- Word2vec Word embedding layer

Keywords:
- RNN
- Deep learning
- BiGRU
- BiLSTM
- Natural language process

ABSTRACT:

Automatically analysing the sentiment of comments expressed by a user on a web page for any purpose is a rapidly expanding important research area. Text sentiment analysis, as it is known in the literature, is a technique that allows users to determine their emotional tendencies in comments defined for any purpose. Users comment on the content of web pages used by thousands of people such as vacation sites, shopping pages, social media, brand reviews, financial reviews, health sites, political pages. The comments made have the ability to directly affect a user who wants to benefit from these services in any way. For these reasons, it is important to examine people's emotions in their comments in automatic review of comments. Recurrent Neural Network (RNN) based architectures have achieved remarkable success in solving Natural Language Processing (NLP) problems. In this article, an RNN based deep learning model is proposed that works on a publicly available dataset obtained from the TripAdvisor web page and performs sentiment analysis. The proposed SAHRAN model uses an attention mechanism based on the dot product structure to capture emotional words in user comments. In the model, Bidirectional Gated Recurrent Unit (BiGRU) and Bidirectional Long Short Term Memory (BiLSTM) deep learning layers are integrated into the model to capture emotional features. As a result of the experimental studies, the proposed SAHRAN model achieved performance values of 0.9524, 0.9685, 0.9082 and 0.9338 in terms of precision, recall, F1 score and accuracy performance measures, respectively.

[1] Halit ÇETİNER (**Orcid ID:** 0000-0001-7794-2555), Isparta University of Applied Sciences, Vocational School of Technical Sciences, Department of Computer Technologies, Isparta,Türkiye
[2] Sedat METLEK (**Orcid ID:** 0000-0002-0393-9908), Burdur Mehmet Akif Ersoy University, Technical Sciences Vocational School, Department of Mechatronics, Burdur, Türkiye

***Sorumlu Yazar/*Corresponding Author*:** Halit ÇETİNER, e-mail: halitcetiner@isparta.edu.tr

INTRODUCTION

Natural language processing (NLP), a sub-branch of artificial intelligence, is a field that transforms important raw data such as speech, text, subtitles, and audio into meaningful data by enabling interaction (Balyan et al., 2020). Due to the development of the digital world, a large amount of raw text data is produced today. According to the study conducted by Hunsinger, it is reported that nearly 20 billion unprocessed raw text data messages are sent every day (Peslak et al., 2018). According to statistics determined by Shiau et al., it is reported that 41,000 searches are made every second with Google, which is among the world's major search engines. In addition, it is reported that 350 million images are uploaded daily with Facebook, one of the popular social networking sites (Shiau et al., 2018).

The Internet has transformed from a stable one-way information movement into a dynamic interactive area where multi-way information movement takes place (Cheng et al., 2021). To analyse the increasing raw information in the internet space, the sentiments in user comments can be automatically analysed to help understand customer dynamics. It is not possible to manually process, analyse, summarize and evaluate thousands of comments in raw texts. However, sentiment analysis can be performed on raw texts with an algorithm that can be implemented with recurrent neural networks. Emotional analysis, summarization and evaluation of raw texts are popular research areas.

Sentiment analysis of raw texts is usually performed with classical methods or machine learning-based methods. Although the mentioned methods have high performance values, they face many difficulties such as creating a dictionary and formulating judgment rules (Nasukawa & Yi, 2003). As the processed words grow and new words come into the processed word pool, the difficulty in establishing semantic connections becomes unbearable (Zhang et al., 2022). Machine learning-based methods that have difficulty distinguishing the semantics of sentences consisting of words can make large errors in sentiment classification (Xiao et al., 2018). Machine learning methods known as bag-of-words models ignore the order of words, causing machine learning-based models to fail to capture context information correctly (Cheng et al., 2021).

Sentiment analysis on the text obtained after some preprocessing performed on raw texts is generally performed in three different categories. The first category is the sentiment analysis method based on the sentiment dictionary. In this method, it divides the pre-processed text into words. Then, it scores according to the weights of some sentiment words. Then, it scores the sentiment calculation of the entire pre-processed text. According to the scoring result, it obtains the classification result. Here, a deficiency in the sentiment dictionary causes major errors. The second category is a sentiment analysis method based on machine learning algorithms (Zhang et al., 2019). The third category is sentiment analysis methods based on deep learning architectures. Instead of extracting too many features, these methods obtain highly distinctive features and enable context learning by obtaining short text-based features. They ensure that text contexts are not forgotten with long-memory algorithms such as Long Short Term Memory (LSTM). In this respect, they provide a better performance result than the other two methods.

In recent years, there has been a significant increase in raw data and significant progress in emotion research (Li et al., 2024). It is seen that approaches based on deep learning-based architectures are being developed to perform emotion classification (Çetiner, 2022). In the basic sentiment classification approaches in the literature, the text is converted into a matrix. It has been determined that convolution operations are then applied to obtain the feature values (Kishwar & Zafar, 2023; Metlek & Çetiner, 2024). A comparison of deep learning-based sentiment analysis methods such as convolutional methods and recurrent neural networks with classical methods shows that deep learning-based methods have superior performance. As a result of the conclusions stated in this paper, BiLSTM and BiGRU based methods are used to identify emotional features in raw texts more comprehensively and at a higher level, and to analyse and process local and global features correctly.

Sentiment analysis in raw texts is a set of techniques developed to identify emotional phrases in texts. Sentiment analysis is used in brand reputation management processes to increase the sustainability of brands, controls in risky areas such as politics, automated stock and fund tracking (Li et al., 2024). Although there are many different techniques developed in this field, challenges and limitations remain. In sentiment analysis, raw data needs to be refined and processed. Semantic variation in raw texts, ambiguity in text data, errors, problems, and unprocessed noise degrade the performance and efficiency of sentiment analysis (Li et al., 2024).

Many natural language experts have conducted research on sentiment analysis. This study examined deep learning-based sentiment analysis approaches. In recent years, researchers have been using deep neural networks extensively in natural language processing problems such as sentiment analysis (Cheng et al., 2021). Compared to classical machine learning techniques that rely on a large number of manual features and deep learning-based approaches, recurrent neural networks are preferred because they are bidirectional and can memorize word contexts for a long time. Attentional mechanisms can effectively select the important texts in the text in sentiment analysis (Vaswani et al., 2017; Zhao & Wu, 2016). Cheng et al. developed a hierarchical-based Chinese text sentiment analysis method with the advantages of convolution-based deep learning layers (Cheng et al., 2019). In their study, they report that RNN-based text analysis studies provide longer text dependency compared to CNN-based studies. They also state that RNN-based methods have a larger memory capability. In addition, RNN-based methods require shorter training time than CNN-based sentiment analysis methods. Cho et al. developed a GRU-based model that provides faster learning with fewer parameters than LSTM architectural models with the ability to capture globally significant features in word contexts (Cho et al., 2014). Cheng et al. extracted local and global features of texts using CNN and BiGRU based architectures to perform text sentiment analysis (Cheng et al., 2021). The extracted features were used for multi-level feature representation with global average pooling layers. As a result of these operations, they performed sentiment analysis on Chinese and English data. Dai and Wu used a feature extraction method based on CNN and BiGRU model and connected with a fully connected layer containing the semantic relationship between the front and back of the texts (Dai et al., 2021). Then, softmax classification activation was used for classification. Zulqarnain et al. developed a two state GRU model based on recurrent neural networks to capture word features (Zulqarnain et al., 2024). The developed model relates the complex connection between sentences and words. It then focuses on capturing emotion keywords with an attention-based approach. Salmony et al. use recurrent neural networks to perform sentiment analysis (Salmony et al., 2023). The study conducted to determine the effects of different word embedding layers on sentiment analysis is a comparison article.

In addition to the studies mentioned above, Başarslan et al. conducted sentiment analysis using the dataset used in the article. In the study, it is seen that they used the term frequency-inverse document frequency (TF-IDF) statistical method and Word2Vec word embedding method to perform word vectorization. In addition, they performed sentiment analysis using the heterogeneous ensemble method based on deep learning methods such as LSTM, RNN, as well as machine learning methods such as decision trees and support vector machines from the machine learning methods of the vectorized words. In their studies, it was deemed insufficient to use only the TF-IDF statistical method and Word2Vec word embedding method. Therefore, in addition to these, word embedding methods called Bag of words (BOW) and Bidirectional Encoder Representations from Transformers (BERT) were also used in our study. Apart from this, the LSTM and RNN models they based their studies on were also the source of inspiration for our study (Başarslan & Kayaalp, 2024).

In recent years, there has been a focus on studies based on attention mechanisms. There is a need for new approaches to sentiment analysis by quickly learning the complex connections between words and sentences.

The main contributions of this article to the approaches in the literature are presented below.

- Focused on the words that contribute the most to sentiment analysis with the Proposed SAHRAN model approach to capture the important relationships between word contexts.
- LSTM and GRU networks were used to improve the representation between word contexts. GRU architecture provided improvement in recording the relationship between word contexts due to the lower number of parameters and gates compared to LSTM networks.
- Although it is recommended in the literature to develop a GRU-based model that provides faster learning with fewer parameters compared to LSTM architectural models, it has been observed that the performance results of both architectural models are close.

In experimental studies conducted with and without the attention mechanism developed to better capture the relationship between word contexts, it was observed that the performance difference between them was very close.

The next steps of the article are organized as follows. In the second section, the sentiment analysis approaches of machine learning and deep learning based methods widely used in the literature are examined in detail. In the third section, information is given about the raw dataset processed in the article. Then, the method used in the article and the proposed sentiment classification algorithm are introduced. In the fourth section, the results of the methods used in terms of precision, recall, F1 score and accuracy performance measures are shared. These results are presented both with tables and graphs. In the fifth section, the article is concluded with ideas that provide hope for future sentiment analysis studies.

MATERIALS AND METHODS

Material

The sentiment classification dataset performed within the scope of this article consists of comments obtained from Tripadvisor.com. Alem et al. who prepared this dataset, examine the important useful shares related to a product and service in the comments of users in their studies (Alam et al., 2016). This dataset has over 860 million unique comments.

Used Word Embedding Techniques

One of the ways to represent word vectors in texts is the word2vec word embedding method published by Google (Dai et al., 2021). Word2vec is a word model based on the work of Mikolov et al. in 2013 (Mikolov, Chen, et al., 2013). Word2vec models can take large text data as input and the models created are unsupervised. Vector spaces are obtained in the words obtained with the Word2vec structure. Word2vec is called neural networks that map words to relatively high-dimensional vectors by converting them into numerical vectors. The word2vec process, which has two different types, Continuous Bag of Words (CBOW) and Skip Gram, has a high sensitivity value (Bali et al., 2019; Chen et al., 2017).

The architecture that uses the surrounding words to predict the words in the center is called CBOW. The architecture that predicts the words around the center word is called Skip Gram architecture. The main reason why the Skip Gram architecture model is preferred is because it produces more successful results than the CBOW model (Karcioğlu & Aydin, 2019).

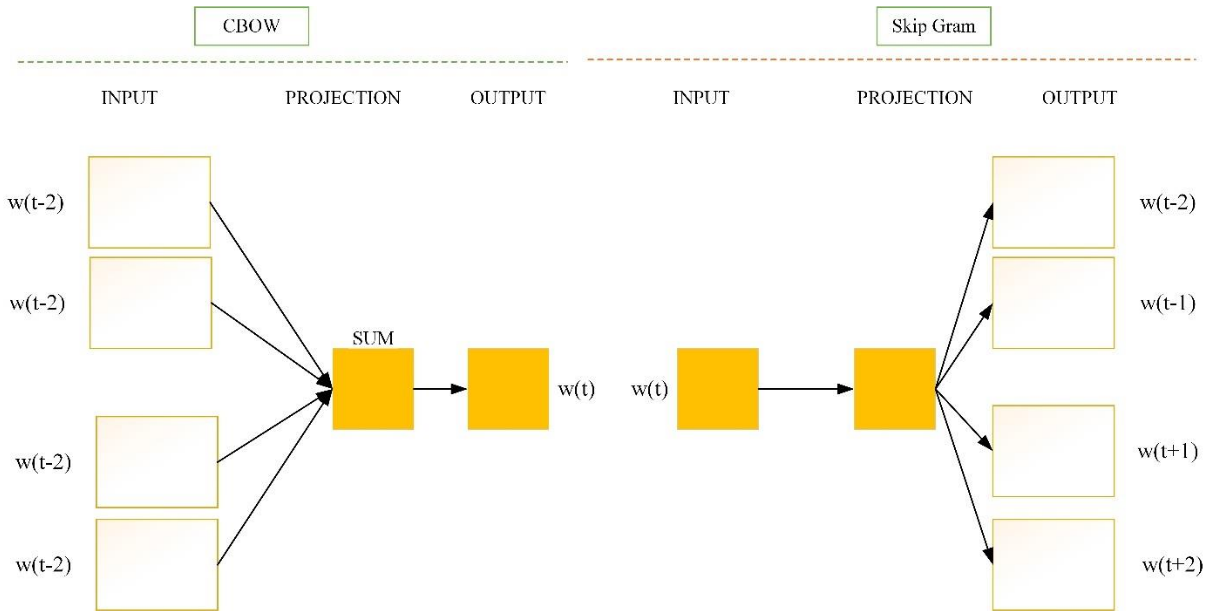

Figure 1. Types of Word2Vec model architecture (Mikolov et al., 2013)

In this study, the Skip Gram model was preferred because it can capture the meaningful relationship between words and produce more successful results. In the Skip Gram model, words near the word of interest receive less attention than words far from it (Çetiner, 2022). As a result of this situation, distant words are assigned less weight value than nearby words.

In some studies, in the literature, instead of the word embedding technique, TF-IDF technique, known as a statistical measure that expresses the importance of a word for a document in a collection, is used. In the TF-IDF technique, the frequency of a word in the target document is very important. The higher the frequency of a word in the target document, the higher its importance. The BOW model also converts text information into numbers like TF-IDF. In the BOW model, the conversion process is carried out by converting into numerical format and looking at the context or order of the words (Aravinthan & Eugene, 2024).

The BERT model is a language model trained on a large dataset whose parameters were adjusted and optimized by Devlin et al. in 2018. The BERT model architecture, based on a multi-layer bidirectional transform encoder, is a model developed by Google for natural language processing (Devlin, 2018).

LSTM

Unlike RNN networks, the advanced neural network model that can hold input information in memory for a long time is called LSTM. It is actively used in natural language processing problems because it can remember input values by keeping them in memory for a long time (Karcıoğlu et al., 2021). LSTM solves the gradient vanishing problem by using sigmoid or tanh activation function that includes input, forget, output gates as well as cell state.

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (1)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (2)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (3)$$
$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (4)$$
$$h_t = o_t \odot tanh(c_t) \quad (5)$$

Equation 1, Equation 2, Equation 3, Equation 4, Equation 5 equations f_t, o_t, c_t, i_t and h_t represent forget, output gates, cell state, input gate and hidden state values respectively. The input vector at time

t is represented by x_t and the symbol σ represents the sigmoid activation function. The W and b parameters in Equations 1-4 represent the weight matrix and the deviation vector (Karcıoğlu et al., 2021).

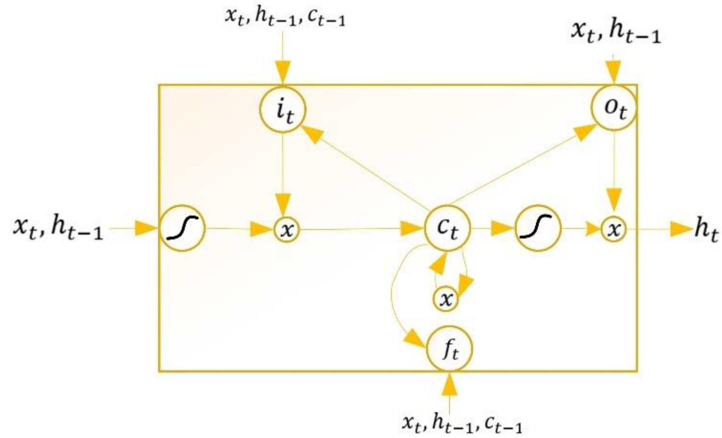

Figure 2. Network model of LSTM architecture (Graves, 2013)

The general LSTM architecture network model shown in Figure 2 shows control gates that can overcome the problems experienced in the RNN architecture. The relevant figure shows the use of input, forget, output gates, as well as cell state and activation functions (Karcıoğlu et al., 2021).

Bidirectional LSTM and GRU

An extension of LSTM architectural models that connects two separate hidden LSTM layers to the same output in opposite directions is called BiLSTM (Hamoud et al., 2022). In BiLSTM architecture, the h_t symbol, which carries information about the past hidden state in the LSTM architecture, can also have future information (Ma, 2016). The hidden states of the forward layer and the hidden states of the backward layer at time t are shown in Equation 6 and Equation 7, respectively. In Equation 9, the forward and backward hidden states are combined with the σ function.

$$\vec{h}_t = \sigma\left(W_{\vec{h}} x_t + W_{\vec{h}\vec{h}} \vec{h}_{t-1} + b_{\vec{h}}\right) \tag{6}$$

$$\overleftarrow{h}_t = \sigma\left(W_{\overleftarrow{h}} x_t + W_{\overleftarrow{h}\overleftarrow{h}} \overleftarrow{h}_{t-1} + b_{\overleftarrow{h}}\right) \tag{7}$$

$$y_t = W_{\vec{h}y} \vec{h}_t + W_{\overleftarrow{h}y} \overleftarrow{h}_t + b_y \tag{8}$$

$$y_t = \sigma\left[\vec{h}_t, \overleftarrow{h}_t\right] \tag{9}$$

RNN-based models are used as the basis in this article because they are more successful in sorted data sets. In text processing applications where the input depth increases, processing time increases (H. Çetiner, 2024a). In RNN-based operations, LSTM and GRU-based models are used instead of standard RNN to eliminate memory loss in short-term operations.

BiGRU is a refresh neural network architecture that is capable of processing texts by taking into account the contexts ahead and behind the point of data processing. BiGRU combines forward and backward GRU units into one GRU unit (Wang et al., 2018). It is similar to the LSTM architecture, but has fewer gates and complexity. It is an easier architecture to use among RNN architectures. In GRU architectures, a hidden layer is used to transmit information between nodes (Çetiner, 2024a).

The update and reset gate in GRU structures can be expressed by the equations between Equations 10 and 13.

$$\tilde{h}_t = \tanh\left(W_h x_t + r_t \odot (U_h h_{t-1})\right) + b_h \tag{10}$$

$$(r_t) = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{11}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{12}$$

$$(z_t) = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{13}$$

While $z_t$in Equation 13 represents the update gate, r_t represents the classification gate. W in Equations 10, 11 and 13 represents the weight value. h represents the values of the hidden layer at the current time. U, σ and x represent the cell units, sigmoid activation function and inputs to the model, respectively. It is an appropriate architecture not only for short-term data series operations but also for capturing long-term dependencies. It contains vector sets containing different gates such as input, forget, output and memory at different time steps. Since there is no possibility of forward and backward navigation from the current point in standard LSTM models, there may be disconnections in the connections. For this reason, LSTM and GRU architectures that provide forward and navigation opportunities are taken as basis.

Proposed SAHRAN Model

In this section, the steps of the algorithm that performs the sentiment analysis process, which is considered as an NLP problem, are discussed in detail (Wankhade et al., 2024). The SAHRAN model proposed in the article consists of three different stages: data preprocessing, word embedding, and deep learning layers with attention mechanism.

There are characters such as spelling mistakes, special characters, non-standard characters in the comments obtained from Tripadvisor.com. The dataset filled with such structures needs to be transformed into a structure that can perform effective sentiment analysis. Therefore, the data preprocessing steps of the proposed SAHRAN model for sentiment analysis are presented in Figure 3.

- In the first pre-processing step in Figure 3, reviews from Tripadvisor.com are retrieved.
- In the pre-processing in the second step of Figure 3, all words are brought to the same standard and spaces are removed. All letters are converted to lower case so that all expressions in the texts are of the same standard. Otherwise, it is possible that the algorithm processes lower and upper case letters differently due to different ASCII code values. In addition to, abbreviations such as ‘4ever’ and words with extended suffixes such as ‘yesss’ were converted to their original formats as ‘forever’ and ‘yes’, respectively.
- In the preprocessing in the third step of Figure 3, special characters, punctuation marks and numbers are removed from the dataset because these special characters, especially numeric characters, do not affect whether a sentence has positive or negative content in sentiment analysis.
- In the preprocessing in the fourth step of Figure 3, URL links and stop words that may be present in text sentences are removed as they are ineffective in sentiment analysis.
- In the pre-processing in the fifth step of Figure 3, comments and sentiment labels values are separated after the text normalization process.
- In the preprocessing in the sixth step of Figure 3, performing tokenization operation on allocated test data.
- In the preprocessing in the seventh step of Figure 3, performing a sequence operation on separated training and test data.
- In preprocessing step eight of Figure 3, create pad sequence with maximum length in training and test data.

  In the pre-processing in the last step of Figure 3, pre-processing is defined with the input layer defined as maximum length.

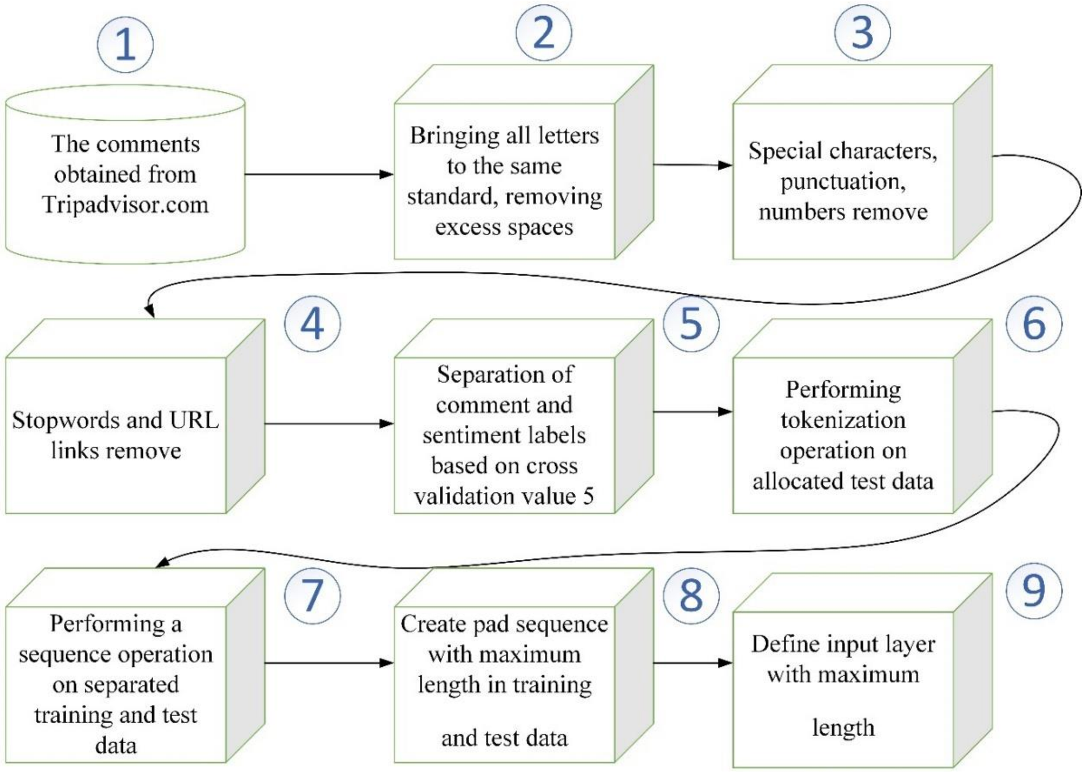

Figure 3. The preprocessing steps of the proposed SAHRAN model

The words in the text to which pre-processing is applied must be expressed numerically (Çetiner, 2022). In this sense, the Word2Vec word embedding method was used with the aim of providing a good result in prediction with the skip gram technique. The words between the windows before and after the current word were tried to be predicted with the windowing process. In the skip gram technique, the words are weighted according to their proximity and distance to the training sample.

$$Q = Cx\left(D + Dxlog_2(V)\right) \tag{14}$$

The complexity of the skip gram technique is calculated according to the structure given in Equation 14. Q in Equation 14 shows the activation output. V, D and C in Equation 14 show the vocabulary size, input size and maximum distance between words, respectively.

In RNN-based algorithms that have difficulty in retaining information from distant sequences, it is necessary to use the attention mechanism to eliminate this problem. The attention mechanism allows examining previous words and deriving from these words in RNN-based models that have difficulty in retaining information from distant information. In sentiment analysis, it is possible to evaluate all past situations using important indicators. The attention mechanism implemented in this article is based on the dot product structure (Luong et al., 2015). It is aimed to strengthen the model prediction by focusing on the important basic elements between the parts of the text input. The proposed SAHRAN model includes the previous time contexts in the calculation by calculating the dot product between the input and the output. Then, the model applies a weighting to measure the input and output.

$$(h_t, x_t) = h_t * \mathrm{W} * x_t \tag{15}$$

In Equation 15, the output, input and trainable matrix values at time t are represented by h_t, x_t and W, respectively. Equation 15 shows the calculated score value.

$$alpha_t = softmax(h_t) \tag{16}$$

$$c_t = alpha_t * \ x_t \tag{17}$$

$$Attention\ result = tanh\left(\mathrm{W} * (c_t + h_t)\right) \tag{18}$$

Equation 16 shows the softmax activation function applied to obtain the attention weight. Equation 17 shows the product of the multiplied input and attention weight to obtain the context vector. Equation 18 shows the tanh activation passed through the dense layer together with the attention output.

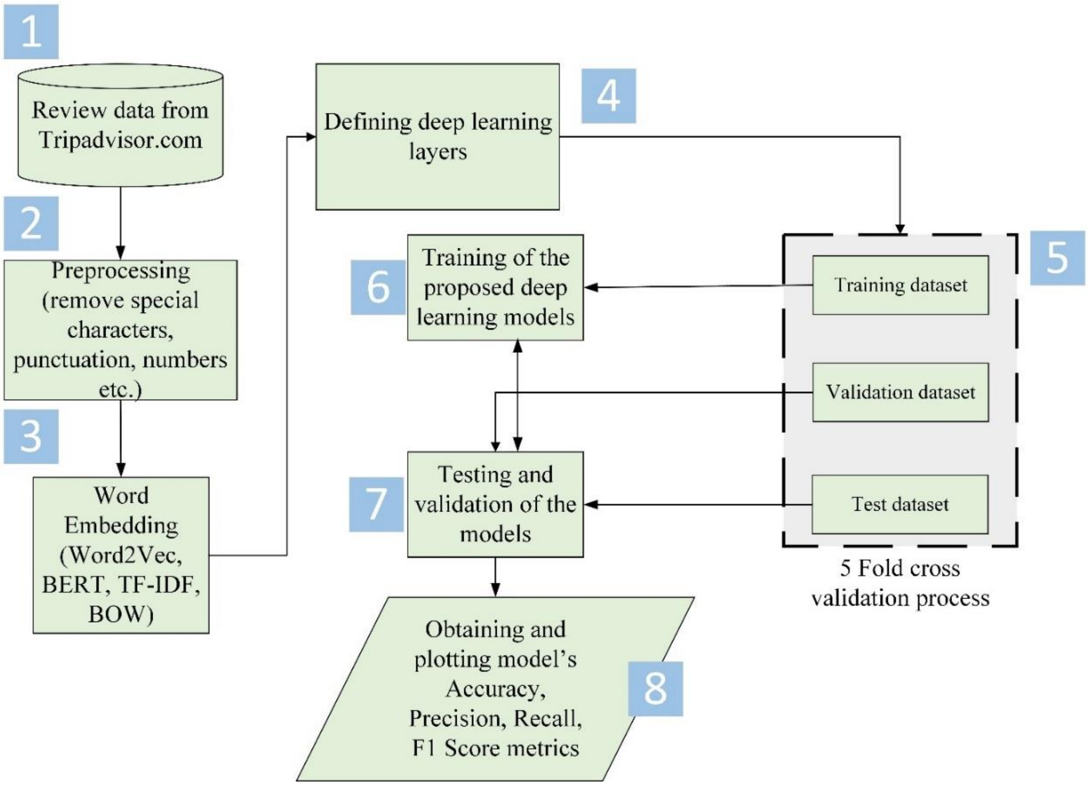

Figure 4. The framework of the proposed SAHRAN model

Figure 4 presents the general framework of the proposed SAHRAN model. The presented model consists of 8 general steps. The first step is to take the publicly available dataset used in sentiment analysis as input. In the second step, all the preprocessing steps shown in Figure 3 are applied to the data to ensure that the data is at the same standard in training, testing and validation processes. In preprocessing, numerical and special characters in the input texts are removed as they do not have an effect on sentiment analysis. Unnecessary spaces on the right and left sides of the texts are removed and the open forms of all abbreviations in the texts are replaced. In addition, all texts are converted to lowercase letters and the texts in the dataset are adjusted to the same standard. Not having the same standard in the dataset can reduce performance and cause unexpected errors (Çetiner, 2022). In the third step, in order to effectively measure the performance of the proposed SAHRAN model, methods widely used in the literature, namely BERT, TF-IDF, BOW, Word2Vec, are used. The performance results obtained in each of these methods are shared in detail in the following sections. In the fourth section, after the input, word embedding layer was prepared, the deep learning layers and parameters suitable for these layers were defined. In the fifth step, in order to ensure the reliability of the prepared model named SAHRAN, the data taken as input was divided into training and test validation groups with the cross validation 5 technique. In the sixth and seventh steps, training and testing processes were performed. In the last step, performance metrics were obtained.

The general framework of the proposed SAHRAN model

The general framework structure of the proposed SAHRAN model is presented in Figure 5. This structure consists of three structures. The first structure is the preprocessing steps specified in Figure 3 after receiving the input values. The second step includes the application of TF-IDF, BOW, BERT and

Word2Vec word embedding techniques. Each of them was tested separately to measure their effect on the proposed SAHRAN model. In the third stage, the deep learning layers shown in Figure 5 were defined and implemented.

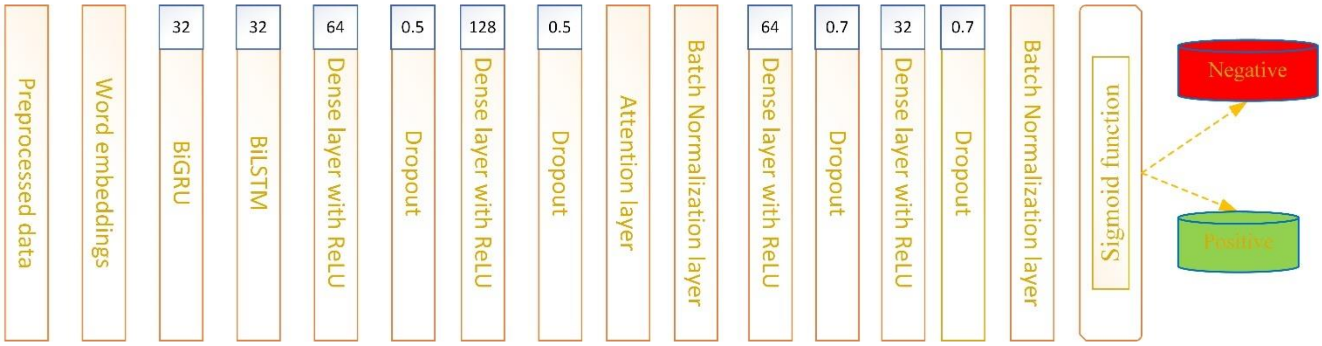

Figure 5. The general framework of the proposed SAHRAN model

The algorithm presented in Figure 5 consists of 18 steps, excluding the input layer and preprocessing steps. In the first step, the data received in the input layer of the algorithm are converted to vectors with word embedding methods. They are given as input to the deep learning layers. In the second and third steps, bidirectional 32-neuron BiGRU and BiLSTM layers are applied one after the other. In the fourth step, the connection is strengthened with a 64-neuron Dense layer with ReLU activation function. In the fifth step, a dropout layer that drops neurons at a rate of 0.5 is defined to prevent over-learning. In the sixth step, the connection is strengthened again with a 128-neuron ReLU activation Dense layer. In the seventh step, another dropout layer is defined that drops neurons at a rate of 0.5.

In the eighth step, the attention layer, which allows the input data to focus on important words, is defined using Equations 15, 16, 17 and 18. In the ninth step, the batch normalization layer, which provides normalization between layers, is added. In the tenth step, the Dense layer, which performs reinforcement with a 64-neuron ReLU activation function, is defined. In the eleventh and thirteenth steps, the dropout layer, which performs neuron dropout at a rate of 0.7, is applied. In the twelfth step, the connections are re-reinforced with the Dense layer with a 32-neuron ReLU activation. In the fourteenth step, the batch normalization layer, which provides normalization between layers, is applied. In the fifteenth step, probabilistic prediction values are obtained with the classification layer with a sigmoid activation function. The highest result is accepted as the class value. In the sixteenth step, the model created using the Adam optimization technique with the binary_crossentropy method is compiled. In the seventeenth step, the model created with a batch size of 32 with a 10 epochs value is trained. In the last step, the performance results of the proposed SAHRAN model were obtained in terms of both training and validation accuracy, precision, recall, and F1 score metrics.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{19}$$

$$Recall = \frac{TP}{TP + FN} \tag{20}$$

$$Precision = \frac{TP}{TP + FP} \tag{21}$$

$$F1\ score = 2x \frac{PrecisionxRecall}{Precision + Recall} \tag{22}$$

In the evaluation of the proposed SAHRAN model, measurement formulas belonging to accuracy, precision, recall and F1 score metrics, which are frequently used in the literature, were used (Karcioğlu et al., 2020, 2021; Çetiner, 2024b; Metlek et al., 2024; Çetiner & Metlek, 2024). In Equation 19, the result is calculated by dividing the number of correct sentiment analysis by the number of all samples.

Equation 22 is a metric that provides balance between Equations 20 and 21. In this equation, the harmonic mean prevents the emergence of exceptions (Karcioğlu & Yaşa, 2020). Anything done to increase Equation 20 will decrease the performance result obtained from Equation 21. Whether there is a balance between precision and recall can be understood by examining the result of Equation 22.

RESULTS AND DISCUSSION

In this section, the performance outputs of the proposed SAHRAN model obtained as a result of the flow diagram defined in Section 3 are presented both in tables and figures. The performance results of the proposed SAHRAN model, which are the measurement metrics commonly used in the literature, such as precision, recall, F1 score and accuracy, are presented in Table 1.

Table 1. Train and validation performance results of the proposed SAHRAN model

Word Embedding Type	Type	Precision	Recall	F1 score	Accuracy
Word2Vec	Train	0.9895	0.9895	0.9094	0.9892
Word2Vec	Validation	0.9524	0.9685	0.9082	0.9338
BERT	Train	0.9991	0.9989	0.9043	0.9984
BERT	Validation	0.9547	0.9561	0.9064	0.9260
TF-IDF	Train	0.9095	0.9715	0.9045	0.9004
TF-IDF	Validation	0.9350	0.9480	0.9064	0.9023
BOW	Train	0.8793	0.9663	0.9045	0.8560
BOW	Validation	0.9049	0.9678	0.9064	0.8890

When the training and validation results in Table 1 are examined, it is seen that both results are close and compatible with each other. The performance measurement results obtained on a class basis are given in Table 2. It can be said that the detection of values with negative classes is more difficult than the detection of texts with positive labels. When the class-based performance results are examined in detail, it is seen that the results of positive classes are better than the results of negative classes. The average accuracy rate obtained with the Word2Vec word embedding method is 93%. The average accuracy rate obtained with the BERT method is 92%. The average accuracy rate obtained with the TF-IDF method is 90%. The average accuracy rate obtained with the BOW method is 89%.

Table 2. Performance results of the class-based proposed SAHRAN model

Word Embedding Type	Classes	Precision	Recall	F1 score
Word2Vec	Negative	0.86	0.72	0.78
Word2Vec	Positive	0.94	0.97	0.96
		Accuracy		0.93
BERT	Negative	0.82	0.74	0.78
BERT	Positive	0.96	0.96	0.95
		Accuracy		0.92
TF-IDF	Negative	0.75	0.69	0.72
TF-IDF	Positive	0.93	0.95	0.94
		Accuracy		0.90
BOW	Negative	0.83	0.52	0.64
BOW	Positive	0.90	0.98	0.94
		Accuracy		0.89

Table 3 shows the confusion matrix results of the proposed SAHRAN model. These results are consistent with the results in Table 1 and Table 2. The confusion matrix results obtained by using the Word2Vec, BERT, TF-IDF and BOW methods given in Table 1 and Table 2 as word embedding methods in the proposed SAHRAN model are presented in Table 3.

Table 3. Confusion matrix results of the proposed SAHRAN model

Word Embedding Type	Classes	Positive	Negative
Word2Vec	Positive	245	96
Word2Vec	Negative	39	1451
BERT	Positive	254	87
BERT	Negative	56	1434
TF-IDF	Positive	234	107
TF-IDF	Negative	77	1413
BOW	Positive	176	165
BOW	Negative	37	1453

Figure 6, Figure 7, Figure 8, Figure 9 and Figure 10 present the results of the performance metrics loss, accuracy, F1 score, recall and precision, respectively. In the relevant figures, the performance results of the Word2Vec and BERT word embedding types that gave the best performance in Table 1 are shared graphically. The training and test results of each word embedding method are shown in different colors.

In Figure 6, the loss information of the performance values is plotted. While the training loss value of the Word2Vec type model is 0.002, the validation loss value reaches 0.44. While the training loss of the BERT model is 0.005, the validation loss reaches 0.44. In both types, the SAHRAN model gives similar results. Although the model needs to be improved, the performance result is satisfactory.

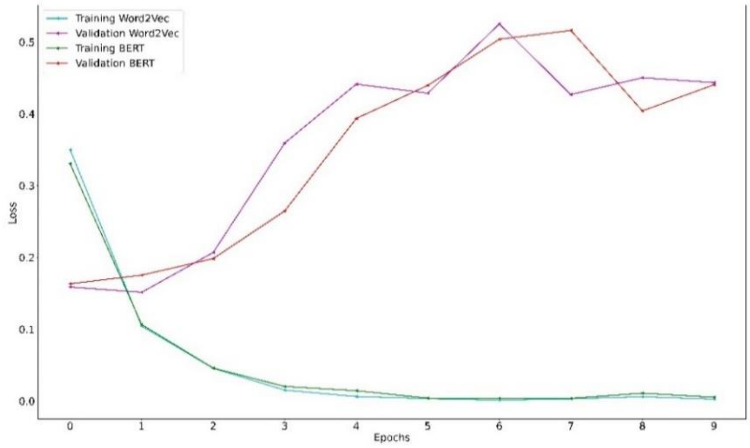

Figure 6. Loss performance results of the proposed SAHRAN model

In Figure 7, the accuracy information of the performances values is plotted. While the training accuracy value of the Word2Vec type model is 0.9892, the validation accuracy value reaches 0.9338. While the training accuracy of the BERT model is 0.9984, the validation accuracy reaches 0.9260. In both types, the SAHRAN model gives similar results.

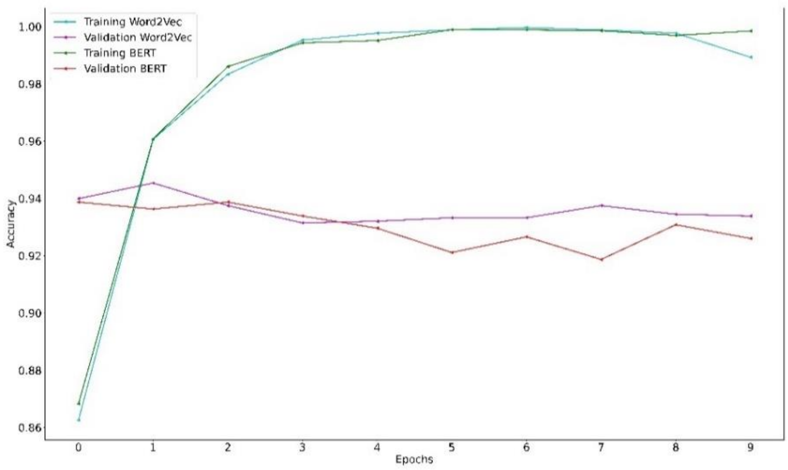

Figure 7. Accuracy performance results of the proposed SAHRAN model

In Figure 8, the F1 score information of the performances values is plotted. While the training F1 score value of the Word2Vec type model is 0.9094, the validation accuracy value reaches 0.9082. While the training F1 score of the BERT model is 0.9043, the validation F1 score reaches 0.9064. In both types, the SAHRAN model gives similar results. However, the model implemented using the Word2Vec type provided a more fluctuating F1 score performance result.

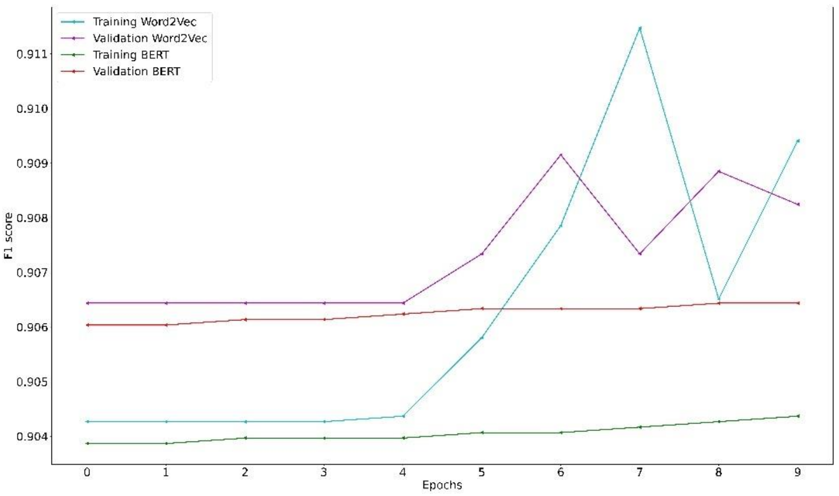

Figure 8. F1 score performance results of the proposed SAHRAN model

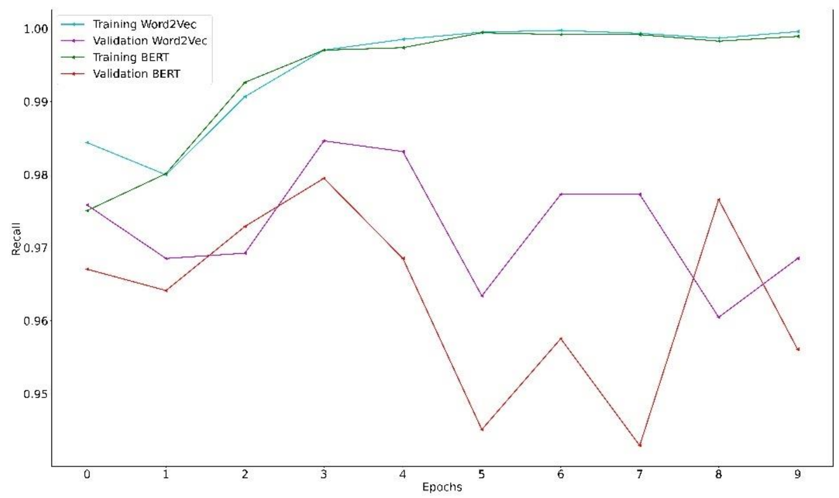

Figure 9. Recall performance result of the proposed SAHRAN model

In Figure 9, the recall information of the performances values is plotted. While the training recall value of the Word2Vec type model is 0.9895, the validation recall value reaches 0.9685. While the training recall of the BERT model is 0.9989, the validation recall reaches 0.9561. In both types, the SAHRAN model gives similar results.

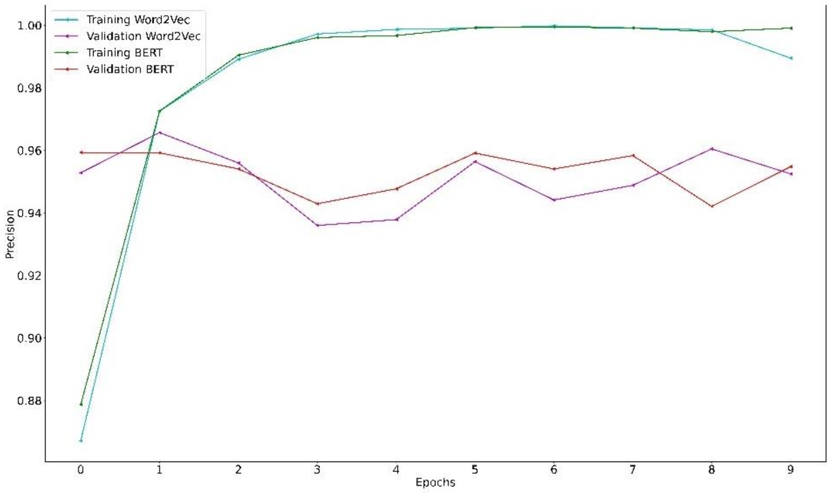

Figure 10. Precision performance result of the proposed SAHRAN model

In Figure 10, the precision information of the performances values is plotted. While the training precision value of the Word2Vec type model is 0.9895, the validation precision value reaches 0.9524. While the training precision of the BERT model is 0.9991, the validation precision reaches 0.9547. In both types, the SAHRAN model gives similar results. While the proposed SAHRAN model provides good performance with both word embedding methods, it is observed that it does not give the same performance when other word embedding methods are used. In order to achieve similar performance over all word embedding methods, an intermediate method can be developed that allows selecting features from all word embedding methods.

Table 4. Performance comparison of the proposed SAHRAN model with studies in the literature using the same dataset

Model & Reference	Dataset & Reference	Precision	Recall	F1 score	Accuracy
LSTM (Priya & Deepalakshmi, 2023)	Hotel Review (Alam et al., 2016)	-	-	-	0.90
RNN (Dang et al., 2020)	Sentiment140 (Go et al., 2009)	0.7773	0.7773	0.6404	0.5695
	Tweets Airline (Eight, 2019)	0.8366	0.9741	0.9001	0.8280
	Twitter SemEval (Evaluation, 2017)	0.5883	0.0946	0.1375	0.5485
	Book Reviews (Blitzer et al., 2007)	0.5614	0.6304	0.5116	0.5169
	Music Reviews (Blitzer et al., 2007)	0.4606	0.7420	0.5673	0.5170
CNN (Dang et al., 2020)	Sentiment140 (Go et al., 2009)	0.7407	0.7407	0.7593	0.7668
	Tweets Airline (Eight, 2019)	0.8366	0.9700	0.9138	0.8545
	Twitter SemEval (Evaluation, 2017)	0.8159	0.7744	0.7943	0.8137
	Book Reviews (Blitzer et al., 2007)	0.7264	0.7300	0.7275	0.7274
	Music Reviews (Blitzer et al., 2007)	0.6912	0.6970	0.6912	0.6920
DNN (Dang et al., 2020)	Sentiment140 (Go et al., 2009)	0.7577	0.7577	0.7638	0.7649
	Tweets Airline (Eight, 2019)	0.8845	0.9556	0.9186	0.8593
	Twitter SemEval (Evaluation, 2017)	0.8350	0.8081	0.8211	0.8367
	Book Reviews (Blitzer et al., 2007)	0.7707	0.7422	0.7550	0.7587
	Music Reviews (Blitzer et al., 2007)	0.7709	0.7650	0.7677	0.7685
CNN (Yildirim, 2022)	TripAdvisor-Hotel Review (Alam et al., 2016)	0.622	0.324	-	0.825
LSTM (Yildirim, 2022)	TripAdvisor-Hotel Review (Alam et al., 2016)	0.050	0.003	-	0.800
RNN (Yildirim, 2022)	TripAdvisor-Hotel Review (Alam et al., 2016)	0.594	0.184	-	0.812
NB (Mostafa, 2020)	TripAdvisor-Hotel Review (Alam et al., 2016)	0.7600	0.6700	-	0.8500
BiLSTM (Zhou, 2019)	TripAdvisor-Hotel Review (Alam et al., 2016)	-	-	-	0.7373
Proposed SAHRAN	TripAdvisor-Hotel Review (Alam et al., 2016)	0.9524	0.9685	0.9082	0.9338

The dataset used in this article is found in very few recent studies in the literature. The same dataset used in this study was also used in the study conducted by (Yildirim, 2022). The results obtained in this study and the results of studies obtained from similar datasets in the literature are shared. It is observed that the precision and recall values in this study are quite low. The proposed method is competitive with the study of Yildirim (2022) using the same dataset in the literature.

In addition to the studies given in Table 4, the performance results of the method proposed in the article were also compared with the Başarslan & Kayaalp (2024)'s study published in recent years.

However, Başarslan's performance results are shared here because they are only accuracy results. The same dataset used in this study was used in the study conducted by Başarslan et al. (Başarslan & Kayaalp, 2024). Başarslan et al. obtained an accuracy rate between 0.870 and 0.898 from the TripAdvisor dataset in their study using the TF-IDF statistical method and the Word2Vec word embedding technique. As a result of the study, they carried out with the help of majority vote of LSTM, RNN, LSTM-RNN architectures and Naïve Bayes (NB), Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT) methods, they achieved a good performance result. TF-IDF and Word2Vec word embedding techniques were used in the study, which also inspired the proposed SAHRAN model study.

In Zhou's study, the highest performance result using the GloVe word embedding technique was 0.7373, which is shown in Table 4 (Zhou, 2019). Mostafa achieved the highest emotion classification result among NB, SVM and DT from the NB classifier (Mostafa, 2020). Articles from recent years using the Tripadvisor database have been scanned. When the performance results in the scanned studies are examined, the performance result obtained in the study is also sufficient.

The datasets used for comparison, apart from the Tripadvisor dataset are Sentiment140, Tweets Airline, Twitter SemEval, Book Review, and Music Reviews. Sentiment140 is a dataset obtained from Stanford University and contains 1.6 million tweets (Go et al., 2009). Tweets Airline contains 14,460 user opinions about an airline (Eight, 2019). Tweets SemEval consists of a set of 17,750 tweets covering a number of geopolitical entities (Evaluation, 2017). Book Reviews and Music reviews is a dataset obtained by Johns Hopkins University (Blitzer et al., 2007). The book and music dataset contains 2000 examples with negative and positive labels as book and music. When the examples of the same and different datasets given in Table 4 are examined, it is seen that the proposed SAHRAN model is at a level that can compete with the studies in the literature.

CONCLUSION

Algorithms based on RNN architecture have recently been widely used in the literature for NLP problems. GRU and LSTM based architectures have been implemented bilaterally in order to eliminate the problem of not being able to contextually capture remote information, which is frequently encountered in RNN based architectural models. BiGRU and BiLSTM based layers are integrated with the attention mechanism and performance evaluation is performed. In this paper, a new approach is provided to replace the analyses performed without focusing on sentiment words in text analysis applications. In the new approach, both word embedding layer and attention mechanism are used. Thanks to the mentioned structures, word weighting is performed and important words are emphasised with certain criteria. Further studies can be carried out by comparing Word2Vec word embedding technique with different word embedding techniques. At the same time, the effect of different attention mechanisms on model performance can be determined in further studies. In addition to these, different studies with high social contribution value and automatic text analysis of user comments can be conducted with detailed analyses on different datasets similar to Tripadvisor.

Conflict of Interest

The article authors declare that there is no conflict of interest between them.

Author's Contributions

The authors declare that they have contributed equally to the article.

REFERENCES

Al Hamoud, A., Hoenig, A., & Roy, K. (2022). Sentence subjectivity analysis of a political and ideological debate dataset using LSTM and BiLSTM with attention and GRU models. Journal of King Saud University - Computer and Information Sciences, 34(10), 7974–7987.

Alam, M. H., Ryu, W.-J., & Lee, S. (2016). Joint multi-grain topic sentiment: modeling semantic aspects for online reviews. Information Sciences, 339, 206–223.

Aravinthan, A., & Eugene, C. (2024). Exploring Recent NLP Advances for Tamil: Word Vectors and Hybrid Deep Learning Architectures. *The International Journal on Advances in ICT for Emerging Regions*, 17(2).

Bali, A. P. S., Fernandes, M., Choubey, S., & Goel, M. (2019). *Comparative Performance of Machine Learning Algorithms for Fake News Detection BT - Advances in Computing and Data Sciences*. Springer Singapore.

Balyan, R., McCarthy, K. S., & McNamara, D. S. (2020). Applying natural language processing and hierarchical machine learning approaches to text difficulty classification. *International Journal of Artificial Intelligence in Education*, 30(3), 337–370.

Başarslan, M. S., & Kayaalp, F. (2024). Sentiment analysis using a deep ensemble learning model. *Multimedia Tools and Applications*, 83(14), 42207–42231.

Blitzer, J., Dredze, M., & Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 440–447.

Çetiner, H. (2022). Multi-Label Text Analysis with a CNN and LSTM Based Hybrid Deep Learning Model. *Journal of Engineering Science of Adiyaman University*, 9(17), 15–16.

Çetiner, H. (2024a). Fake News Detection and Classification with Recurrent Neural Network Based Deep Learning Approaches. *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 7(3), 973–993.

Çetiner, H. (2024b). Skin Lesions Identification and Analysis with Deep Learning Model Using Transfer Learning. *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 7(3), 1030–1045.

Çetiner, H., & Metlek, S. (2024). Analysis of Different Pooling Functions on A Convolution Neural Network Based Model. *International Journal of 3D Printing Technologies and Digital Industry*, 8(2), 266–276.

Çetiner, M. (2022). *Analysis of sustainable fashion products using a deep learning approach* (PhD thesis). Süleyman Demirel University, Faculty of Business Administration.

Chen, T., Xu, R., He, Y., & Wang, X. (2017). Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Systems with Applications*, 72, 221–230.

Cheng, Y., Sun, H., Chen, H., Li, M., Cai, Y., Cai, Z., & Huang, J. (2021). Sentiment Analysis Using Multi-Head Attention Capsules With Multi-Channel CNN and Bidirectional GRU. *IEEE Access*, 9, 60383–60395.

Cheng, Y., Ye, Z. M., Wang, M. W., Zhang, Q., & Zhang, G. H. (2019). Analysis of Chinese text sentiment orientation based on convolutional neural network and hierarchical attention network. *The Chinese Journal of Process Engineering*, 33(1), 133–142.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *ArXiv Preprint ArXiv:1406.1078*.

Dai, Y., Wu, Z., & Zhang, H. (2021). Sentiment Analysis of Comment Texts Based on CNN-BiGRU-Attention. *2021 China Automation Congress (CAC)*, 2749–2754.

Dang, N. C., Moreno-García, M. N., & De la Prieta, F. (2020). Sentiment Analysis Based on Deep Learning: A Comparative Study. In *Electronic*, 9(3).

Devlin, J. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv Preprint ArXiv:1810.04805*.

Eight, F. (2019). *Twitter Airline Sentiment*. URL: https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment (accessed date: March 24, 2024).

Evaluation, S. (2017). *SemEval-2017*. URL: https://alt.qcri.org/semeval2017/ (accessed date: March 24, 2024).

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(12), 2009.

Graves, A. (2013). Generating sequences with recurrent neural networks. *ArXiv Preprint ArXiv:1308.0850*.

Karcioğlu, A. A., & Aydin, T. (2019). Sentiment Analysis of Turkish and English Twitter Feeds Using Word2Vec Model. *2019 27th Signal Processing and Communications Applications Conference (SIU)*, 1–4.

Karcioğlu, A. A., & Bulut, H. (2021). Performance Evaluation of Classification Algorithms Using Hyperparameter Optimization. *2021 6th International Conference on Computer Science and Engineering (UBMK)*, 354–358.

Karcioğlu, A. A., & Yaşa, A. C. (2020). Automatic Summary Extraction in Texts Using Genetic Algorithms. *2020 28th Signal Processing and Communications Applications Conference (SIU)*, 1–4.

Karcıoğlu, A. A., Tanışman, S., & Bulut, H. (2021). Türkiye'de COVID-19 Bulaşısının ARIMA Modeli ve LSTM Ağı Kullanılarak Zaman Serisi Tahmini. *Avrupa Bilim ve Teknoloji Dergisi*, *32*, 288–297.

Kishwar, A., & Zafar, A. (2023). Fake news detection on Pakistani news using machine learning and deep learning. *Expert Systems with Applications*, 211, 118558.

Li, D., Shi, X., & Dai, M. (2024). *A Text Sentiment Classification Method Enhanced by Bi-GRU and Attention Mechanism BT-Proceedings of the 13th International Conference on Computer Engineering and Networks*. Springer Nature Singapore.

Luong, T., Pham, H., & Manning, C. D. (2015). Effective Approaches to Attention-based Neural Machine Translation. In L. Màrquez, C. Callison-Burch, & J. Su (Eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1412–1421). Association for Computational Linguistics.

Ma, X. (2016). End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *ArXiv Preprint ArXiv:1603.01354*.

Metlek, S., & Çetiner, H. (2024). Inception SH: A New CNN Model Based on Inception Module for Classifying Scene Images. *Mühendislik Bilimleri ve Tasarım Dergisi*, 12(2), 328–344.

Mikolov, T., Chen, K., Corrado, G. ., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR*, 2013.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 3111–3119.

Mostafa, L. (2020). Machine learning-based sentiment analysis for analyzing the travelers reviews on Egyptian hotels. *The International Conference on Artificial Intelligence and Computer Vision*, 405–413.

Nasukawa, T., & Yi, J. (2003). Sentiment analysis: Capturing favorability using natural language processing. *Proceedings of the 2nd International Conference on Knowledge Capture*, 70–77.

Peslak, A., Hunsinger, S., & Kruck, S. (2018). Text messaging today: A longitudinal study of variables influencing text messaging from 2009 to 2016. *Journal of Information Systems Applied Research*, 11(3), 25.

Priya, C. S. R., & Deepalakshmi, P. (2023). Sentiment analysis from unstructured hotel reviews data in social network using deep learning techniques. *International Journal of Information Technology*, 15(7), 3563–3574.

Salmony, M. Y., Faridi, A. R., & Masood, F. (2023). Leveraging attention layer in improving deep learning models performance for sentiment analysis. *International Journal of Information Technology*.

Shiau, W.-L., Dwivedi, Y. K., & Lai, H.-H. (2018). Examining the core knowledge on facebook. *International Journal of Information Management*, 43, 52–63.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need.(Nips), 2017. *ArXiv Preprint ArXiv:1706.03762*, 10, S0140525X16001837.

Wang, Y., Liao, W., & Chang, Y. (2018). Gated Recurrent Unit Network-Based Short-Term Photovoltaic Forecasting. In *Energies*, 11(8).

Wankhade, M., Annavarapu, C. S. R., & Abraham, A. (2024). CBMAFM: CNN-BiLSTM Multi-Attention Fusion Mechanism for sentiment classification. *Multimedia Tools and Applications*, 83(17), 51755–51786.

Xiao, Z., Li, X., Wang, L., Yang, Q., Du, J., & Sangaiah, A. K. (2018). Using convolution control block for Chinese sentiment analysis. *Journal of Parallel and Distributed Computing*, 116, 18–26.

Yildirim, G. (2022). A novel grid-based many-objective swarm intelligence approach for sentiment analysis in social media. *Neurocomputing*, 503, 173–188. https://doi.org/10.1016/j.neucom.2022.06.092.

Zhang, M., Zheng, R., Chen, J., Zhu, J., Liu, R., Sun, S., & Wu, Q. (2019). Emotional component analysis and forecast public opinion on micro-blog posts based on maximum entropy model. *Cluster Computing*, 22, 6295–6304.

Zhang, W., Li, X., & Liu, W. (2022). A Review of Text Sentiment Analysis. *International Journal of Social Science and Education Research*, 5(9), 23–28.

Zhao, Z., & Wu, Y. (2016). Attention-Based Convolutional Neural Networks for Sentence Classification. *Interspeech*, 8, 705–709.

Zhou, Y. (2019). *Sentiment classification with deep neural networks* (Master of Science Thesis). Tampere University, Faculty of Information Technology and Communication Sciences.

Zulqarnain, M., Ghazali, R., Aamir, M., & Hassim, Y. M. M. (2024). An efficient two-state GRU based on feature attention mechanism for sentiment analysis. *Multimedia Tools and Applications*, 83(1), 3085–3110