Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3840071, 10 pages
https://doi.org/10.1155/2022/3840071

Research Article

Developing an Intelligent System with Deep Learning Algorithms for Sentiment Analysis of E-Commerce Product Reviews

Mohammad Eid Alzahrani,[1] Theyazn H. H. Aldhyani ,[2] Saleh Nagi Alsubari,[3] Maha M. Althobaiti ,[4] and Adil Fahad[1]

[1] *Department of Engineering and Computer Science, Al Baha University, Al Bahah, Saudi Arabia*
[2] *Applied College in Abqaiq, King Faisal University, P.O. Box 400, Al-Ahsa 31982, Saudi Arabia*
[3] *Department of Computer Science & Information Technology, Dr. Babasaheb Amedkar Marathwada University, Aurangabad, India*
[4] *Department of Computer Science, College of Computing and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia*

Correspondence should be addressed to Theyazn H. H. Aldhyani; taldhyani@kfu.edu.sa

Received 26 February 2022; Revised 1 May 2022; Accepted 11 May 2022; Published 28 May 2022

Academic Editor: Muhammad Ahmad

Most consumers rely on online reviews when deciding to purchase e-commerce services or products. Unfortunately, the main problem of these reviews, which is not completely tackled, is the existence of deceptive reviews. The novelty of the proposed system is the application of opinion mining on consumers' reviews to help businesses and organizations continually improve their market strategies and obtain an in-depth analysis of the consumers' opinions regarding their products and brands. In this paper, the long short-term memory (LSTM) and deep learning convolutional neural network integrated with LSTM (CNN-LSTM) models were used for sentiment analysis of reviews in the e-commerce domain. The system was tested and evaluated by using real-time data that included reviews of cameras, laptops, mobile phones, tablets, televisions, and video surveillance products from the Amazon website. Data preprocessing steps, such as lowercase processing, stopword removal, punctuation removal, and tokenization, were used for data cleaning. The clean data were processed with the LSTM and CNN-LSTM models for the detection and classification of the consumers' sentiment into positive or negative. The LSTM and CNN-LSTM algorithms achieved an accuracy of 94% and 91%, respectively. We conclude that the deep learning techniques applied here provide optimal results for the classification of the customers' sentiment toward the products.

1. Introduction

Web 3.0 has the main features of the semantic web, artificial intelligence, connectivity, etc., allowing people to use social media to communicate and express their opinions about real-world events. In this context, the analysis of users' reviews is essential for companies to grow worldwide. This makes opinion mining a key player in the analysis of reviews and discussions. Nowadays, companies analyze this type of information to improve the quality and performance of the products and, consequently, survive in a competitive market. Opinion mining can be expressed as the reason behind any action or movement that people use to follow the reason [1].

Within the huge amount of data generated on the Internet, important information is hidden. Data mining techniques are used to extract information and solve various problems. Online product reviews have two important aspects under which data are stored on the Internet. Commercial websites are platforms where users express their sentiment or opinion on several topics. Sentiment analysis refers to a broad area of natural language processing (NLP), computational linguistics, and text mining [2]. The use of these techniques leads to the extraction and analysis of the opinion on a given product. Opinion mining defines an opinion as positive or negative, and sentiment analysis defines the polarity value of a user's opinion on a particular product or service. The current approaches of sentiment analysis are mainly [3] machine learning algorithms [4], lexicon-based methods, [5] and hybrid models [6, 7].

Negation is a prevalent morphological development that impacts polarity and, therefore, must be reflected in the assessment of sentiment. Automatic detection of negation in news articles is required for numerous text processing applications, including sentiment analysis. Here, we explored the role and importance of users' reviews concerning particular products on the decision using sentiment analysis. We present experimental results that demonstrate that sentiment analysis is appropriate to this end. The goal was to determine the polarity of the natural language of texts written in product reviews.

The existing straightforward approaches are statistical, based on frequencies of positive and negative words. Recently, researchers discovered new ways to account for other aspects of content, such as structural or semantic features. The present work focuses on the identification of document-level negation by using multiple computational methods. In recent years, with the exponential growth of smartphone use, many people are connected to social networking platforms, like Facebook, Twitter, and Instagram. Social networks have become a field to express beliefs or opinions, emotions, thoughts, personal issues, places, or personalities.

There are numerous studies applying sentiment analysis, some of which used real-time data from Twitter for extracting patterns by employing the Twitter-streaming application programming interface (API) [8, 9]. The sentiment analyzers are divided into two types: SentiWordNet [10] and WordNet [11]. Sentiment analysis uses positive and negative scores to classify opinions. By developing a model to analyze word sequence disambiguation [12], the Twitter-streaming API was used to gather data concerning the Indonesian presidential elections [13]. Needless tweets were removed, and the remaining data were investigated for sentimental aspects by dividing each tweet into numerous sub-tweets and calculating the sentiment polarity of the sub-tweets for predicting the consequence of the elections. The mean absolute error metric was used to evaluate the results, it noted that the prediction error was 0.6 better than the previous study [14]. To predict the Swedish election outcome by using Twitter data, a system was developed [15]. To predict the outcome of the European elections, a new method was designed that studied the similarity of the structure with the outcome of the vote. Another method was created to test Brazilian municipal elections in six cities [16]. In this methodology, sentiment analysis was applied along with a stratified sample [17] of users to compare the characteristics of the findings with the actual voters.

Many researchers have used machine learning and artificial intelligence to analyze the sentiment of tweets [18, 19]. In [20], the Naive Bayes, support vector machine (SVM) [21], and information entropy-based [22] models were applied to classify product reviews. A hybrid machine learning algorithm based on Twitter opinion mining was proposed in [23]. Heydari et al. [24] proposed time series model for fraudulent sentiment reviewer analysis. Hajek et al. [25] developed a deep feedforward neural network and convolution model to detect fake positive and negative review in an Amazon dataset. Long et al. [26] applied LSTM with multi-head attention network for predicting sentiment-based text using China social media dataset. Dong et al. [27] proposed supervised machine linear regression for predicting sentiment of customers presented in online shopping data using sentiment analysis learning approaches.

Researchers have been focusing on developing powerful models to deal with the ever-increasing complexity of big data [28, 29], as well as expanding sentiment analysis to a wide range of applications [30, 31], from financial forecasting to marketing strategies [32] among other areas [33, 34]. However, only a few of them analyzed different deep learning approaches to give real evidence of their performance [35]. Deep learning techniques are becoming increasingly popular. When assessing the performance of a single approach on a single dataset in a specific area, the results suggest that CNN and RNN have relatively good accuracy. Based on AdaBoost combination, Gao et al. [36] proposed CNN model for sentiment analysis in user-generated text. In this vein, Hassan and Mahmood [37] demonstrated that the CNN and RNN models overcame the problem of short texts in deep learning models.

Some traditional approaches, which are assisted by machine learning techniques, are based on aspects of the used language. Using the domain of movie opinions, Pang et al. [18] studied the performance of various machine learning algorithms, including Naive Bayes, maximum entropy, and SVM. By using SVM with unigrams, they achieved an accuracy of 82.9%. NLP is typically used to extract features used by a sentiment classifier. In this aspect, the majority of NLP strategies are centered on the usage of n-grams but the use of a bag-of-words strategy is also common [38, 39]. Numerous studies have demonstrated significant results when employing the bag-of-words as a text representation for item categorization [40–44].

Researchers have taken advantage of NLP themes to develop deep learning models based on neural networks with more than three layers, according to the journal Nature. Most of these studies found that deep learning models accurately detect sentiment in various situations. The CNN [45, 46], RNN [47], deep neural network [48], recursive neural deep model [49], and the attention-based bidirectional CNN-RNN [50] models are some representative examples. Some researchers combine models, which are then referred to as hybrid neural networks. The hierarchical bidirectional RNN is an example of a hybrid neural network [51]. The main issue with sentiment analysis of product reviews in the e-commerce domain is the existence of fake reviews that lead customers to select undesired products [52].

The main contributions of the proposed research are the following:

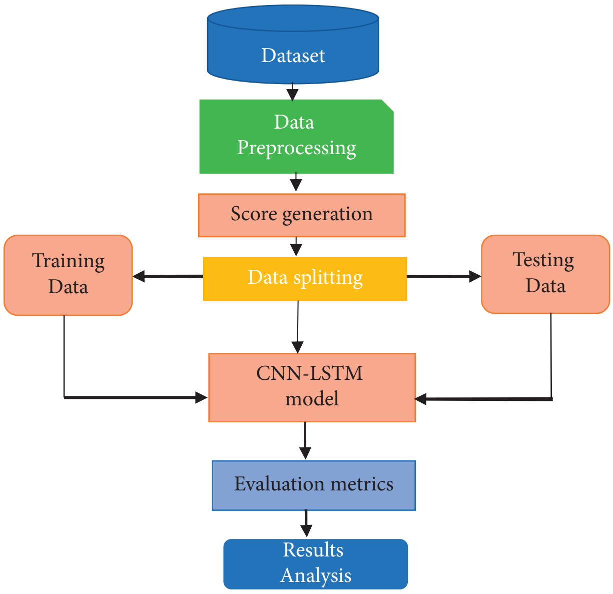


Figure 1: Framework for the proposed methodology.

(1) The generation of a sentiment score using a lexicon-based approach for each product review of the dataset.

(2) Labeling the review texts as negative if the generated sentiment score is <0 or positive if the score is >1.

(3) The combination of all product reviews into a single data frame to obtain more sentiment-related words.

(4) Improving the accuracy by developing a hybrid deep learning model combining the CNN and LSTM models for the product-related sentiment classification.

(5) Comparing the classification performance of the CNN-LSTM and LSTM models.

2. Materials and Methods

The proposed methodology for predicting the review-related sentiments is based on the deep learning algorithms presented here. The phases of the proposed system are the following: dataset collection, data preprocessing, generating the sentiment score, polarity calculation, applying the CNN-LSTM model, evaluation metrics, and analysis of the results. Figure 1 shows the framework of the proposed methodology used in the present study.

2.1. Datasets. To evaluate the proposed system, the dataset [53] was collected from reviews on the Amazon website in JSON file format. Each JSON file comprises a number of reviews (Table 1). The dataset includes reviews of laptops, mobile phones, tablets, televisions, and video surveillance products. The data preprocessing includes various steps, such as lowercase processing with meta-features like the reviewer's ID, the product ID, and the review text.

Table 1: The number of reviews per product category.

Product name	Review count
Laptops	1,946
Mobile phones	1,918
Tablets	1,894
Televisions	1,596
Video surveillance products	2,597

2.2. Data Preprocessing. We implemented different preprocessing steps aiming at cleaning the review texts so that they are easy to process. The following preprocessing methods were performed on the dataset as a whole.

2.2.1. Lowercase. It entails converting whole words of the review text into lowercase words.

2.2.2. Stopword Removal. Stopwords are widely used words in a language, such as "the," "a," "an," "is," and "are". As these words do not carry any information significant for the model, they were removed from the content of the review.

2.2.3. Punctuation Removal. All punctuation marks in the review texts were removed.

2.2.4. One-Word Review Elimination. Reviews that included only one word were eliminated.

2.2.5. Contraction Removal. This process replaces a word originally written in the short form with the respective full form; for instance, "when've" becomes "when have."

2.2.6. Tokenization. Each sentence of the review texts was divided into small pieces of words or tokens.

2.2.7. Part-of-Speech Tagging. This step is used to tag each word present in the sentence with a POS tag, for example, "VB" for a verb, "AJJ" for an adjective, and "NN" for a noun.

2.2.8. Score Generation. The review text was evaluated for sentiment, and a score was generated. For calculating the sentiment score, the dataset was matched with opinion lexicon [53] that consists of 5,000 positive words and 4,500 negative words with their respective scores. The sentiment score was calculated for each review text based on the scores of the lexicon. The review text was labeled as positive if the score was >0; otherwise, it was labeled as negative.

2.2.9. Word Embeddings. We calculated numerical vectors with every preprocessed sentence in the product review dataset using the "Word embeddings" method. To create word indices, we first turned all of the review text terms into sequences. The Keras text tokenizer [54] is being used to obtain those indices. We made sure that no term or word gets a zero index in the tokenizer, and that the vocabulary size is adjusted properly. Then, for each single word in the training and testing sets, a distinctive index is generated,

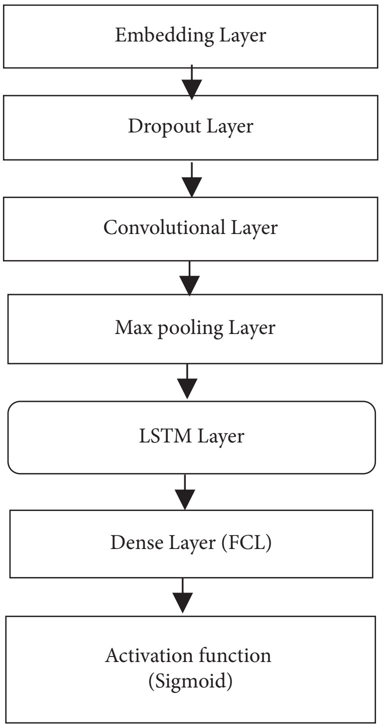


Figure 2: The structure of the CNN-LSTM model.

which is employed to create numeric vectors of all review texts of the dataset.

2.3. The CNN-LSTM Model. Figure 2 presents the structure of the CNN-LSTM model used for sentiment classification of customers' reviews using an Amazon dataset.

2.3.1. Embedding Layer. This is the initial layer of the CNN-LSTM model that is used to transform each word in the training dataset into an actual-valued vector, meaning that a set of sentiment-related words are constructed and transformed into a numerical form. This process is known as word embedding. The embedding layer consisted of three components: the vocabulary size (maximum features; 15,000 words), the embedding dimensions (50), and the input sequence length (400 words).

2.3.2. Dropout Layer. The main task of this layer is to avoid the overfitting of the model [52]. Here, we assigned the value 0.4 to the dropout rate parameter, where this value has a range between 0 and 1. The main function of the dropout layer is to arbitrarily deactivate a set of neurons in the embedding layer, where every neuron denotes the dense exemplification of a sentiment word in a review text.

CNN is a deep learning technique used in different areas such as natural language preprocessing tasks, computer vision, and medical image processing.

2.3.3. Convolution Layer. The third layer of the CNN-LSTM model is used for the extraction of features from the input matrix. It uses n convolution filters that operate over the elements of the input sequence matrix to find the convolutions for each sequence. We set the number of filters to 64 and the size of the filter kernel to 3×3.

2.3.4. Max Pooling Layer. This layer performs down-sampling beside the spatial dimensionality of the given input sequences. It considers the maximum value of all input features in the pool of each filter kernel. It has assigned to 5×5 kernel.

2.3.5. LSTM Layer. LSTM is a type of RNN capable of learning long-term dependence [52]. We used an LSTM layer and assigned it to 50 hidden units toward the next layer. One of the most notable advantages of employing a convolutional neural network as feature extraction technique beyond a traditional LSTM is the reduction in the aggregating amount of features. Throughout the feature extraction process, a sentiment classification model uses these features (words) for prediction of the product review text as positive or negative sentiment. LSTM executes precalculations for the input sequences before providing an output to the last layer of the network. In every cell, four discrete computations are conducted based on four gates: input (i_t), forget (f_t), candidate (c_t), and output (o_t). The structure of the LSTM model is presented in Figure 3. The equations for these gates are as follows:

$$\begin{aligned}
f_t &= \operatorname{sig}\left(W f_{xt} + U f_{ht} - 1 + b_f\right),\\
i_t &= \operatorname{sig}\left(W i_{xt} + U i_{ht} - 1 + b_i\right),\\
O_t &= \operatorname{sig}\left(W o_{xt} + U o_{ht} - 1 + b_o\right),\\
c \sim t &= \tanh\left(w c_{xt} + U c_{ht} - 1 + bc\right),\\
C_t &= \left(f_{to} ct - 1 + i_{to} c \sim t\right),\\
h_t &= O_{to} * \tanh\left(C_t\right),\\
\tanh(x) &= \frac{1 - e^{2x}}{1 - e^{2x}},
\end{aligned} \tag{1}$$

where sig and tanh are the sigmoid and tangent activation functions, respectively, X is the input data, W and b represent the weight and bias factor, respectively, C_t is the cell state, $c \sim t$ is the candidate gate, and h_t refers to the output of the LSTM cell.

2.3.6. Dense Layer (Fully Connected Layer). This is a hidden layer in the CNN-LSTM model. It consists of 512 artificial connected neurons that connect all neurons of the network. The function applied to this layer is the rectified linear unit described by the following equation:

$$f(x) = \max(o, x). \tag{2}$$

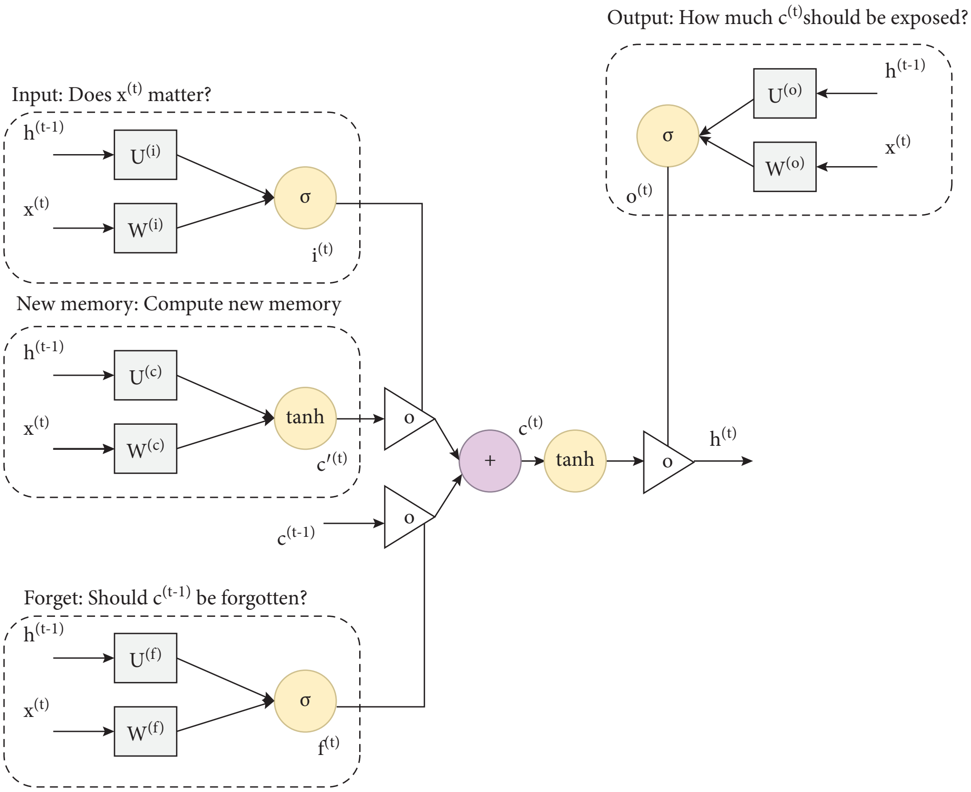


Figure 3: The structure of the LSTM model.

2.3.7. Sigmoid Activation Function. It is the first layer that detects and classifies the output classes (positive or negative sentiment). The sigmoid function formula is given as follows (Algorithm 1):

$$\sigma = \frac{1}{1 - e^{2x}}. \tag{3}$$

2.4. Evaluation Metrics. To evaluate the proposed models (CNN-LSTM and LSTM), the accuracy, precision, recall, F1-score, and specificity metrics were used. The performance measurements are presented below:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{FN} + \text{TP} + \text{TN}} \times 100\%,$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\%,$$

$$\text{F1} - \text{score} = 2 * \frac{\text{precision} \times \text{sensitivity}}{\text{precision} + \text{sensitivity}} \times 100\%, \tag{4}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \times 100\%,$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%,$$

where true positive (TP) represents the total number of samples that are successfully classified as positive sentiment, false positive (FP) is the total number of samples that are incorrectly classified as negative sentiments, true negative (TN) denotes the total number of samples that are successfully classified as negative sentiment, and false negative (FN) represents the total number of samples that are incorrectly classified as positive sentiments.

3. Experimental Results

In this section, we present the experimental results of the application of the CNN-LSTM and LSTM models for the analysis and prediction of sentiment in the e-commerce domain. We used hardware with 4 GB RAM and an i7 2800 CPU and ran the experiments on the Jupyter environment. The evaluation metrics (accuracy, precision, F1-score, recall, and specificity) were employed to examine the proposed system. The word cloud (sentiment words and product names) of the dataset is presented in Figure 4, which shows graphical representations of words (large font words) that give greater importance to that seem more repeatedly in the used product review dataset.

3.1. Data Splitting. In this phase, we divided the dataset that consisted of 13,057 product reviews into 70% training, 10% validation, and 20% testing datasets. Then, the CNN-LSTM and LSTM models were applied to detect and classify the review texts into positive or negative. Table 2 shows the splitting of the dataset.

```
Step 1: input training set as x_train, targets as y_train
Step 2: assign the hyperparameters as embedding_dimension = 50, number_filters = 64, vocabulary_size = 15000 words,
input_length = 400, dropout_rate = 0.4, strides = 5, activation_function = Relu, kernel_size = 3 × 3, pool_size = 5 × ,
lstm_units = 50, batch_size = 32, number_epochs = 5, num_classes = 2, optimizer = (Adam).
Step 3: initialize sequential model ()
Step 4: set embedding layer as input layer
Model = model.add(embedding(embedding_dimension, vocabulary_size, input_length))
Step 5: add convolutional layer
Model = model.add(convolution 1D(number_filters, kernel_size))
Step 6: add max pooling layer
Model = model.add(max_pool layer(pool_size, strides))
Step 7: add LSTM layer
Model = model.add(LSTM_layer(lstm_units, activation_function, recurrent_activation, dropout_rate, return_sequences))
Step 8: add dropout layer
Model-model.add(Dropout(dropout_rate))
Step 9: add dense layer
Model = model.add(Dense_layer(num_classes, activation_function = “sigmoid”))
Step 10: compilation
model.compile(e (loss_function, optimizer)
model.fit (y_train, y_train, number_epochs, batch_size)
```

Algorithm 1: CNN-LSTM.

Figure 4: Word cloud of the dataset.

Table 2: The splitting of the dataset.

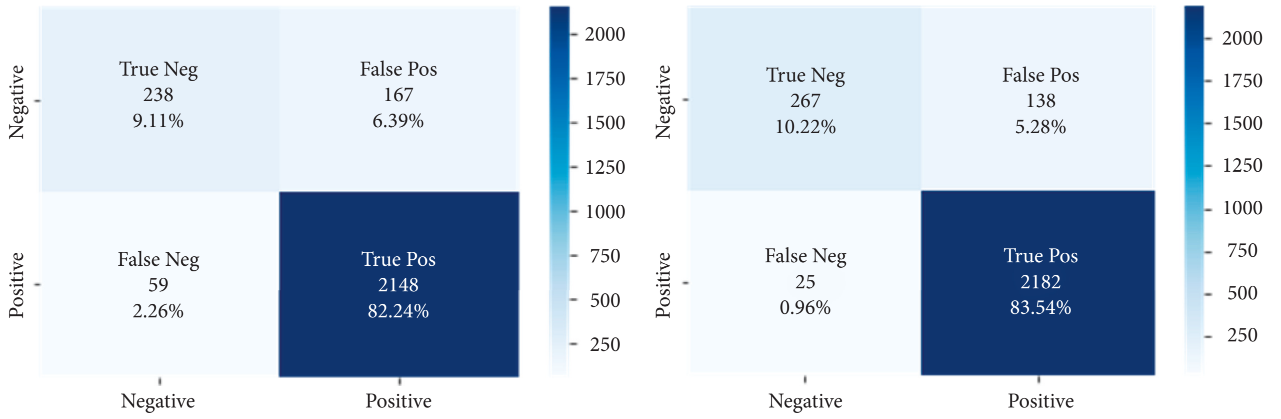
Total number of reviews	Training set 80%	Validation set 10%	Testing set 20%
13,057 (11,184 positive; 1,873 negative)	9,400	1,045	2,612

Table 3: Results of the deep learning models.

Models	Specificity	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
LSTM	95	91.03	92.07	97.73	95.50
CNN-LSTM	96	94	94	99	96.03

3.2. Results and Discussion. Table 3 shows the results of the deep learning approaches. The CNN-LSTM model achieved high accuracy (96%).

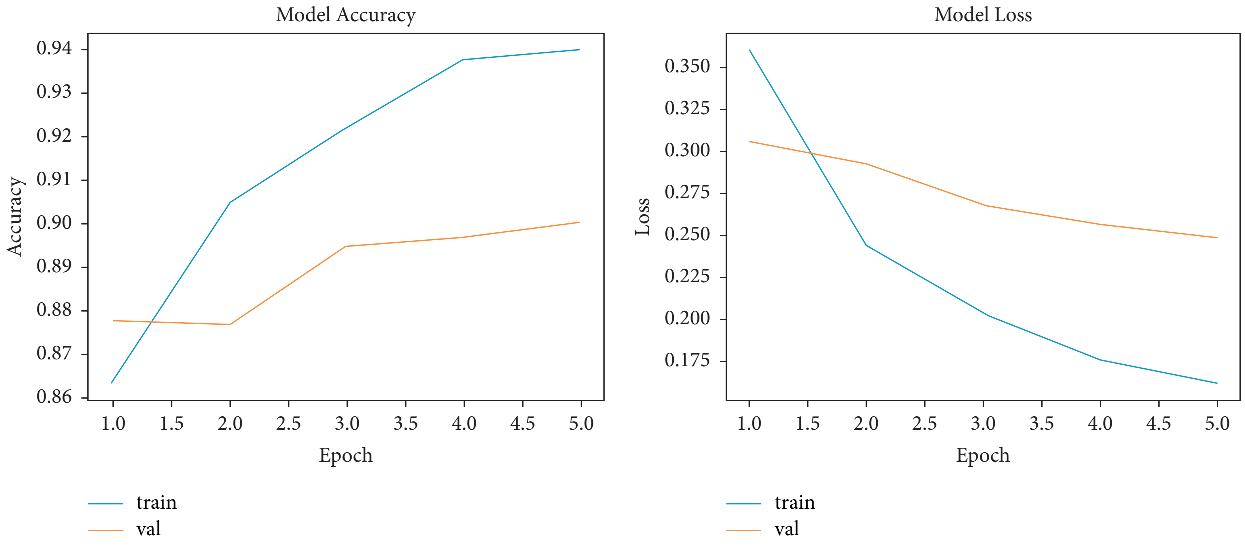
The confusion matrix of the CNN-LSTM and LSTM models is shown in Figure 5. The confusion matrix is used to present the rates of TP, FP, TN, and FN of the sample. Based on these rates, the evaluation metrics (specificity, accuracy, recall, precision, and F1-score) were calculated to evaluate the CNN-LSTM model using unseen data to predict the sentiment of customers. LSTM resulted in 82.24% TP, while



(a)

(b)

Figure 5: Confusion matrix of the (a) LSTM and (b) CNN-LSTM models.



(a)

(b)

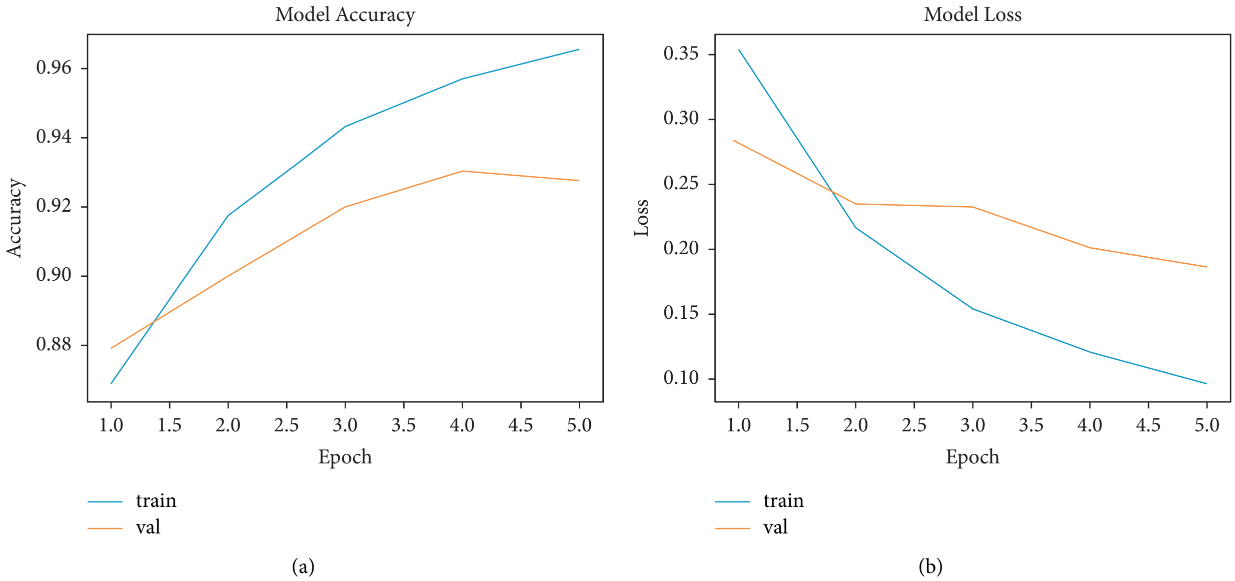
Figure 6: The performance of the LSTM model: (a) accuracy and (b) loss.

CNN-LSTM resulted in 83.54% TP. As for misclassification, LSTM resulted in 6.39% FP and CNN-LSTM in 5.28% FP, indicating that the CNN-LSTM model was slightly better than the LSTM model.

The accuracy performance of LSTM for the training and validation datasets is presented in Figure 6. The LSTM model presented increasing accuracy during the training phase (from 86% to 94%), whereas in the testing phase, it achieved 91% accuracy with 10 epochs. The loss of the LSTM model in the training phase decreased from 5 to 0.35, while in the validation phase, the model loss decreased from 0.3 to 0.27.

The accuracy performance of the CNN-LSTM during the training phase increased from 87.50% to 97%. In the validation phase, the accuracy performance reached 94% (Figure 7(a)). The loss of the CNN-LSTM model in the validation phase was 0.20 (Figure 7(b)).

The dataset developed by Rajkumar et al. [53] proposed SVM and Naive Bayes methods to predict sentiment analysis. They collected data from Amazon concerning mobile phones, tablets, cameras, and televisions. They applied the SVM method to each dataset individually. Here, we applied deep learning models to all the datasets combined. The empirical results of our system were compared with the results of [28] and are shown in Table 4. The CNN-LSTM model achieved an accuracy of 94%.



(a)

(b)

Figure 7: The performance of the CNN-LSTM model: (a) accuracy and (b) loss.

Table 4: Significant results of the CNN-LSTM model compared to the SVM method.

Models	Datasets	Accuracy (%)
Support vector machine [28]	Televisions, tablets, mobile phones, laptops, and video surveillance	88, 84, 92, 88, and 93
Proposed system (CNN-LSTM)	All dataset	94

4. Conclusion

Recently, sentiment analysis has become a valuable tool for the generation and evaluation of different types of data, helping the decision-making processes that lead to the improvement of businesses and companies. Social networking creates a large amount of data that require processing and analysis to obtain relevant insights. In the present study, the experimental dataset was collected from the Amazon website and included reviews of laptops, mobile phones, tablets, televisions, and video surveillance products. The lexicon-based approach was used for the calculation of the sentiment score for each review text. The output of the preprocessed data was classified with the LSTM and CNN-LSTM models. The experimental results showed that our model was satisfactory in all the measurement metrics.

Data Availability

We have collected dataset from authors of research article: https://www.researchgate.net/publication/326985579_Sentiment_Analysis_ on_Product_Reviews_Using_ Machine_Learning_ Techniques_Proceeding_of_CISC_ 2017

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors deeply acknowledge Taif University for supporting this research through Taif University Researchers Supporting Project Number (TURSP-2020/328), Taif University, Taif, Saudi Arabia. This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (Project No: GRANT674).

References

[1] E. Cambria and D. S. A. Das, "Affective computing and sentiment analysis," *A Practical Guide to Sentiment Analysis*, vol. 31, pp. 1–10, 2017.

[2] O. Jagdale, V. Harmalkar, S. Chavan, and N. Sharma, "Twitter mining using R," *Int. J. Eng. Res. Adv. Tech.*vol. 3, pp. 252–256, 2017.

[3] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: a survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014.

[4] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[5] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[6] R. Prabowo and M. Thelwall, "Sentiment analysis: a combined approach," *Journal of Informetrics*, vol. 3, no. 2, pp. 143–157, 2009.

[7] Y. Dang, Y. Zhang, and H. Chen, "A lexicon-enhanced method for sentiment classification: an experiment on online product reviews," *IEEE Intelligent Systems*, vol. 25, no. 4, pp. 46–53, 2010.

[8] R. Jose and V. S. Chooralil, "Prediction of election result by enhanced sentiment analysis on twitter data using word sense disambiguation," in *Proceedings of the 2015 International*

Conference on Control Communication & Computing India (ICCC), pp. 638–641, Trivandrum, India, November 2015.

[9] A. Esuli and F. S. Sebastiani, *A High-Coverage Lexical Resource for Opinion Mining*, Institute of Information Science and Technologies (ISTI) of the Italian National Research Council (CNR), Pisa, Italy, 2006.

[10] G. A. Miller, "WordNet," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[11] R. Navigli, "Word sense disambiguation," *ACM Computing Surveys*, vol. 41, no. 2, pp. 1–69, 2009.

[12] M. Ibrahim, O. Abdillah, A. F. Wicaksono, and M. Adriani, "Buzzer detection and sentiment analysis for predicting presidential election results in a twitter nation," in *Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pp. 1348–1353, Atlantic City, NJ, USA, November 2015.

[13] N. Dokoohaki, F. Zikou, D. Gillblad, and M. Matskin, "Predicting Swedish elections with twitter: a case for stochastic link structure analysis," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1269–1276, Paris, France, August 2015.

[14] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 7, pp. 1019–1031, 2007.

[15] R. Miranda Filho, J. M. Almeida, and G. L. Pappa, "Twitter population sample bias and its impact on predictive outcomes: a case study on elections," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1254–1261, Paris, France, August 2015.

[16] E. Foreman, *Survey Sampling Principles*, CRC Press, Boca Raton, FL, USA, 1991.

[17] S. N. Alsubari, S. N. Deshmukh, A. Abdullah Alqarni et al., "Data analytics for the identification of fake reviews using supervised learning," *Computers, Materials & Continua*, vol. 70, no. 2, pp. 3189–3204, 2022.

[18] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?" in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing—EMNLP '02'*, vol. 10, pp. 79–86, Philadelphia, PA, USA, July 2002.

[19] G. Gautam and D. Yadav, "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," in *Proceedings of the 2014 Seventh International Conference on Contemporary Computing (IC3)*, pp. 437–442, Noida, India, August 2014.

[20] T. Joachims, "Text categorization with support vector machines: learning with many relevant features," in *Proceedings of the 10th European Conference on Machine Learning*, pp. 137–142, Chemnitz, Germany, April 1998.

[21] A. L. Berger, V. J. D. Pietra, and S. A. D. Pietra, "A maximum entropy approach to natural language processing Comput," *Linguist*, vol. 22, pp. 39–71, 1996.

[22] F. H. Khan, S. Bashir, and U. Qamar, "Tom: twitter opinion mining framework using hybrid classification scheme," *Decision Support Systems*, vol. 57, pp. 245–257, 2014.

[23] A. Mukherjee, V. Venkataraman, B. Liu, and N. S. Glance, "What yelp fake review filter might be doing?" in *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, pp. 409–418, Cambridge, MA, USA, July, 2013.

[24] A. Heydari, M. Tavakoli, and N. Salim, "Detection of fake opinions using time series," *Expert Systems with Applications*, vol. 58, pp. 83–92, 2016.

[25] P. Hajek, A. Barushka, and M. Munk, "Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining," *Neural Computing & Applications*, vol. 32, no. 23, Article ID 17259, 2020.

[26] F. Long, K. Zhou, and W. Ou, "Sentiment Analysis of Text Based on Bidirectional LSTM with Multi-Head Attention," *Advanced Optical Imaging for Extreme Environments*, vol. 7, Article ID 141960, 2019.

[27] J. Dong, Y. Chen, A. Gu et al., "Potential Trend for Online Shopping Data Based on the Linear Regression and Sentiment Analysis," *Mathematical Problems in Engineering*, vol. 2020, Article ID 4591260, 11 pages, 2020.

[28] S. S. Roy, M. Biba, R. Kumar, R. Kumar, and P. Samui, "A new SVM method for recognizing polarity of sentiments in twitter," in *Handbook of Research on Soft Computing and Nature-Inspired Algorithms*, pp. 281–291, IGI Global, Hershey, Pennsylvania, 2017.

[29] M. J. S. Keenan, *Advanced Positioning, Flow, and Sentiment Analysis in Commodity Markets*, Wiley, Hoboken, NJ, USA, 2018.

[30] R. Satapathy, E. Cambria, and A. Hussain, *Sentiment Analysis in the Bio-Medical Domain*, Springer, Berlin, Germany, 2017.

[31] A. Rajput, "natural language processing, sentiment analysis, and clinical analytics," in *Innovation in Health Informatics*, pp. 79–97, Elsevier, Amsterdam, The Netherlands, 2020.

[32] J. Qian, Z. Niu, and C. Shi, "Sentiment analysis model on weather related tweets with deep neural network," in *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, pp. 31–35, Macau, China, February 2018.

[33] D.-H. Pham and A.-C. Le, "Learning multiple layers of knowledge representation for aspect based sentiment analysis," *Data & Knowledge Engineering*, vol. 114, pp. 26–39, 2018.

[34] G. Preethi, P. V. Krishna, M. S. Obaidat, V. Saritha, and S. Yenduri, "Application of deep learning to sentiment analysis for recommender system on cloud," in *Proceedings of the 2017 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pp. 93–97, Dalian, China, July 2017.

[35] Q. Tul, M. Ali, A. Riaz et al., "Sentiment analysis using deep learning techniques: a review," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, p. 424, 2017.

[36] Y. Gao, W. Rong, Y. Shen, and Z. Xiong, "Convolutional neural network based sentiment analysis using Adaboost combination," in *Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 1333–1338, Vancouver, BC, Canada, July 2016.

[37] A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in *Proceedings of the Third International Conference on Control, Automation and Robotics (ICCAR)*, pp. 705–710, Nagoya, Japan, April 2017.

[38] M. Kraus and S. Feuerriegel, "Sentiment analysis based on rhetorical structure theory:Learning deep neural networks from discourse trees," *Expert Systems with Applications*, vol. 118, pp. 65–79, 2019.

[39] L. Li, T.-T. Goh, and D. Jin, "How textual quality of online reviews affect classification performance: a case of deep learning sentiment analysis," *Neural Computing & Applications*, vol. 32, no. 9, pp. 4387–4415, 2018.

[40] P. Singhal and P. Bhattacharyya, *Sentiment Analysis and Deep Learning: A Survey*, Center for Indian Language Technology Indian Institute of Technology, Bombay, Indian, 2016.

[41] F. Abid, M. Alam, M. Yasir, and C. Li, "Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter," *Future Generation Computer Systems*, vol. 95, pp. 292–308, 2019.

[42] A. S. M. Alharbi and E. de Doncker, "Twitter sentiment analysis with a deep neural network: an enhanced approach using user behavioral information," *Cognitive Systems Research*, vol. 54, pp. 50–61, 2019.

[43] G. Beigi, X. Hu, R. Maciejewski, and H. Liu, "An overview of sentiment analysis in social media and its applications in disaster relief," in *Sentiment Analysis and Ontology Engineering*, pp. 313–340, Springer, Cham, Switzerland, 2016.

[44] K. Sikka, T. Wu, J. Susskind, and M. Bartlett, "Exploring bag of words architectures in the facial expression domain," in *Proceedings of the European Conference on Computer Vision*, pp. 250–259, Florence, Italy, October 2012.

[45] R. Bekkerman and J. Allan, *Using Bigrams in Text Categorization*, Center of Intelligent Information Retrieval, UMass: Amherst, MA, USA, 2004.

[46] J. Krapac, J. Verbeek, and F. Jurie, "Modeling spatial layout with Fisher vectors for image categorization," in *Proceedings of the International Conference on Computer Vision*, pp. 1487–1494, Barcelona, Spain, November 2011.

[47] X. Wang, A. McCallum, and X. Wei, "Topical n-grams: phrase and topic discovery, with an application to information retrieval," in *Proceedings of the Seventh IEEE International Conference on Data Mining (ICDM)*, pp. 697–702, Omaha, Nebraska, October 2007.

[48] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 959–962, Santiago, Chile, August 2015.

[49] L. Yanmei and C. Yuda, "Research on Chinese micro-blog sentiment analysis based on deep learning," in *Proceedings of the 8th International Symposium on Computational Intelligence and Design (ISCID)*, pp. 358–361, Hangzhou, China, December 2015.

[50] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: an attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Generation Computer Systems*, vol. 115, pp. 279–294, 2021.

[51] L. Arras, G. Montavon, K. R. Müller, and W. Samek, "Explaining Recurrent Neural Network Predictions in Sentiment Analysis,".

[52] S. N. Alsubari, S. N. Deshmukh, M. H. Al-Adhaileh, F. W. Alsaade, and T. H. Aldhyani, "Development of Integrated Neural Network Model for Identification of Fake Reviews in E-Commerce Using Multidomain Datasets," *Applied Bionics and Biomechanics*, vol. 2021, Article ID 5522574, 11 pages, 2021.

[53] R. S. Jagdale, V. S. Shirsat, and S. N. Deshmukh, "Sentiment analysis on product reviews using machine learning techniques. Cognitive informatics and soft computing," *Advances in Intelligent Systems and Computing*, vol. 768, 2018.

[54] Tensorflow and text preprocessing, Accessed 20 January 2022, https://www.tensorflow.org/api_docs/python/tf/keras/preprocess /text/Tokenize.