Neural Computing and Applications
https://doi.org/10.1007/s00521-024-09590-5

ORIGINAL ARTICLE

A novel deep learning model for detection of inconsistency in e-commerce websites

Mohamed A. Kassem[1] · Amr A. Abohany[2] · Amr A. Abd El-Mageed[3] · Khalid M. Hosny[4]

Received: 13 May 2023 / Accepted: 5 February 2024

Abstract
On most e-commerce websites, there are two crucial factors that customers rely on to assess product quality and dependability: customer reviews provided online and related ratings. Reviews offer feedback to customers about the product's merits, reasons for negative reviews, and feelings of satisfaction or dissatisfaction with the provided service. As for ratings, they express customer opinions about the product's quality as numerical values from one to five (one or two for the worst opinion, three for the neutral opinion, and four or five for the best opinion). Usually, the customer reviews may be inconsistent with their relevant ratings; the customer may write the worst review despite providing a four- or five-star rating or write the best review with only a one- or two-star rating. Due to this inconsistency, customers may need help to identify relevant information. Therefore, it is required to develop a model that can classify reviews as either positive or negative, depending on the polarity of thoughts, to demonstrate if there is an inconsistency between customer reviews and their actual ratings by comparing them with the ratings resulting from the model. This paper proposes an efficient deep learning (DL) model for classifying customer reviews and assessing whether there is inconsistency. The recommended model's performance and stability are examined on a large dataset of product reviews from Amazon e-commerce. The experimental findings showed that the proposed model dominates and significantly outperforms its peers regarding prediction accuracy and other performance measures.

Keywords E-commerce · Reviews · Ratings · Inconsistency · Deep learning (DL)

1 Introduction

Communication with customers on e-commerce websites takes place electronically rather than physically. For that, each customer needs to pinpoint precisely the product they desire and then confirm its characteristics by corresponding it with the features shown on the product page. After that, the customer checks reviews from peer customers to see what they said about the product's quality being purchased and provides written justification. Reviews provide much information [1, 2], including user opinions on products, reasons for unfavorable reviews, and recommendations.

✉ Khalid M. Hosny
k_hosny@zu.edu.eg

Mohamed A. Kassem
mohamed.a.kassem@ai.kfs.edu.eg

Amr A. Abohany
cio_kfs@kfs.edu.eg

Amr A. Abd El-Mageed
amr.atef@commerce.sohag.edu.eg

[1] Department of Robotics and Intelligent Machines, Faculty of Artificial Intelligence, Kafrelsheikh University, Kafrelsheikh 33516, Egypt

[2] Faculty of Computers and Informatics, Kafrelsheikh University, Kafrelsheikh, Egypt

[3] Department of Information Systems, Sohag University, Sohâg 82511, Egypt

[4] Department of Information Technology, Faculty of Computers and Informatics, Zagazig University, Zagazig 44519, Egypt

Published online: 16 March 2024

This information reflects satisfaction or dissatisfaction with the presented service and assists customers and e-commerce websites. It enables customers to make well-informed decisions and fosters customer loyalty to e-commerce websites [3, 4]. In contrast, e-commerce website developers concentrate on the customers' core problems, which enhances service goodness and boosts customer satisfaction [5, 6].

Ratings and reviews are consumers' most reliable data sources, providing vital information to help them make more informed decisions before purchasing products. In addition to the reviews, customers may also give star ratings in the form of numeral values, which convey their general opinion of the product's quality. In prevalent practice, these numeral values are typically provided in the range of one to five, where a star rating of one or two represents the worst opinion, and the middle (or neutral) opinion is depicted by a star rating of three. In contrast, a four- or five-star rating declares the best view [7].

The issue arises when there is a discrepancy between customer reviews and the related ratings. Some customer reviews can be fabricated by paying human content producers to write reviews of genuine products. Also, text-generation algorithms can be utilized to provide customer reviews [8]. We may often discover that customer reviews and the corresponding ratings are inconsistent. In contrast, we might find that a customer wrote a highly negative review but gave a four- or five-star rating or a highly positive one- or two-star rating. Competitors might employ this strategy of inconsistency by flooding negative thoughts about their rivals. By over-saturating the market with these reviews, online platforms' ranking algorithms may reduce the attack firm's visibility.

Most e-commerce websites want to address this inconsistency in customer reviews. As a result, this inconsistency may cause customers to find it challenging to find pertinent information among several reviews. It is necessary to develop a model capable of rating reviews according to the classification of review polarity, which might be either positive or negative, to tackle this issue. Hence, the resulting rating review is compared to the actual customer rating review to illustrate whether there is a similarity and consistency between them or whether the customer review rating is inconsistent and unreliable in making decisions before purchasing products.

The bag-of-words or bag-of-n-grams models [9] were utilized to transform customer reviews into a vector with fixed-length features, which were then used for training. These models have significant weaknesses despite their popularity and success. They disregarded the semantic relationship between words, messed up the word order, and had problems with large-dimensional and data variance. DL has recently demonstrated promising sentiment analysis (SA) findings. SA is locating and classifying the customer sentiment (customer satisfaction and dissatisfaction) expressed in a text. Word vector [10] is an unsupervised technique that effectively captures the semantics of words. Words are depicted as vectors, and words with comparable semantic properties are closer to one another in vector space. Linguistic patterns are included in this text's vector representation. Like word vectors, paragraph vector [11] is an unsupervised technique that understands feature vectors with the constant length for text with changing length. In various classification applications, paragraph vector has assured its superiority over bag-of-words models and other types of text representation. It can determine the semantic relationships of a text with high accuracy and requires little training time.

SA, also known as opinion mining, is one of the research topics that are hot and growing, making it challenging to keep up with all the activity there. It aims to investigate how people's opinions and attitudes toward various subjects, events, problems, people, and things are conveyed via text reviews or comments. The biggest online store in the world, Amazon, is an example of one that permits its consumers to rate its products and submit reviews freely. Analyzing these reviews to determine whether they are favorable or harmful will help customers make decisions that range from buying products like cameras and phones to writing reviews for movies and making investments, all of which will have a significant impact.

1.1 Motivation

In the realm of e-commerce, customer reviews and ratings are pivotal in shaping purchase decisions. However, a significant challenge arises due to the inconsistency often observed between the textual reviews and the numerical ratings provided by customers. For instance, a review may express dissatisfaction but be accompanied by a high rating, or vice versa. This discrepancy hinders the ability of potential buyers to make informed decisions as they struggle to ascertain the true sentiment behind a product review. Addressing this challenge, our paper introduces a novel deep learning (DL) model designed to detect and analyze inconsistencies between customer reviews and ratings on e-commerce platforms. The model harnesses the power of two-branch deep learning architecture to efficiently classify customer reviews into positive or negative categories and then assesses the congruency between these classifications and the actual customer ratings. To illustrate the practical challenges our model aims to overcome, consider the following example. A customer leaves a review for a smartphone, stating, "The camera quality exceeded my expectations, capturing stunning images in low light," yet assigns a rating of two stars. Traditional

methods might struggle to interpret this inconsistency, leading to confusion for future customers. Our model is designed to identify such discrepancies, enhancing review systems' reliability and aiding customers in making better purchasing decisions. This paper not only demonstrates the effectiveness of our DL model through extensive experimentation but also highlights its superiority in terms of prediction accuracy and performance measures compared to existing models.

1.2 Contributions

The main contributions of this paper are as follows:

1. Development of an end-to-end deep learning (DL) model for classifying customer reviews and addressing inconsistency in customer review ratings.
2. Assessment of the proposed DL model's effectiveness using a large dataset of Amazon e-commerce product reviews.
3. Conversion of unstructured text into structured format for analysis using the DL model, through various preprocessing operations including data cleaning, normalization, and more.
4. Introduction of a novel two-branch deep learning model, diverging from traditional models like LSTM and RNNs, for classifying customer reviews and acquiring anticipated ratings.
5. Comparison of the DL model's predicted customer review ratings with actual customer review ratings to check for consistency or inconsistency.
6. Execution of several experiments to test the stability and performance of the proposed DL model.
7. Demonstration of the proposed DL model's superiority over other methods in terms of prediction accuracy and various evaluation metrics.

1.3 Paper structure

The remainder of the paper is structured as follows. Section 2 covers the related work and literature review. Section 3 presents the suggested methodology and its components. Section 4 offers the experimental findings and comparisons. Finally, conclusions and suggestions for future work are outlined in Sect. 5.

2 Literature review

This paper is based on two lines of research: firstly, a line that applies sentiment classification and analysis to analyze textual information. Secondly, a line that predicts customer ratings using customer reviews. This section offers some recent studies on these two lines, as follows. Recently, DL has grown in popularity because it can use its hierarchical designs to learn high-level abstractions. DL algorithms have been used in many research to examine online reviews. A recursive neural tensor network was presented in [12] for semantic compositionality over a sentiment tree bank. This model achieved acceptable results for the Stanford sentiment tree bank's single-sentence positive/negative sentiment classification. A convolutional neural network (CNN) [13] with several layers was utilized by Kalchbrenner et al. [14] to get a good SA outcome. By convolutional processes, the method showed various positioning latent, dense, and low-dimensional word vectors. Nonetheless, CNN has proven to be an effective method for classifying text; it ignored sequential information when learning long-distance dependencies while concentrating on local sentence features, significantly impacting classification.

Tang et al. [15] introduced a neural network to generate sentence representations of a document expression consecutively. LSTM networks [16, 17] or CNN are used to learn the model of sentence representation. Then, using a gated recurrent unit (GRU), the sentences' semantics and relationships are encoded into the document expression. In [18], a neural network model was offered that determines the review text's semantics but also the information of the related user. A continuous matrix was employed to represent a user, and a neural network was used to classify sentiments based on the product of a word matrix. A user matrix [19] utilized a matching one-layer CNN to realize IMDB and Yelp datasets review embeddings. The presented CNN used various lengths of reviews to generate 300-dimensional vectors. One-dimensional convolution was executed through moving filters of widths 3 and 5 on the word embeddings and generated numerous feature maps. Only meaningful features were included by applying maximum-overtime pooling in the pooling layer. A 300-dimensional vector was made by concatenating the output of many filters. The Softmax function [20] was employed as an activation function to train the network over K-classes. To anticipate review ratings of restaurant reviews from Yelp and product reviews from Amazon, Seo et al. [21] employed attention-based CNNs.

For classifying the sentiment of product reviews, a DL architecture was presented in [22] that employed widely accessible ratings as weak supervision signals. The framework comprised two stages: knowing a high-level expression that captured the overall sentences' sentiment distribution through rating information. A classification layer was added to the embedding layer, and labeled sentences were used for supervised fine-tuning. To model review phrases, CNN and LSTM network structures were investigated. A dataset from Amazon was used to assess

the architecture. Using customer reviews of hotels on the island of Tenerife, Martin et al. [23] designed and tested classifiers that relied on CNN and LSTM and concluded that LSTM RNNs outperformed CNNs in predicting review ratings. To classify online Amazon Reviews using the Amazon API, the effectiveness of three machine learning techniques—support vector machine (SVM), Naive Bayes (NB), and maximum entropy (ME)—was compared in [24]. The reviews were separated as positive, neutral, and hostile. Uni-grams and weighted uni-grams incorporating positive and negative keywords were employed to extract features and train the presented machine learning classifiers, improving accuracy.

To address the mismatch between the provided Amazon.com customer review and the associated rating, SA utilizing a DL approach was carried out [25]. Paragraph vectors were used first to understand the semantic relationships of a review text by converting the product reviews of Amazon.com to fixed-length feature vectors. Review embedding was also categorized and sorted to create a product sequence supplied to GRU for product embedding recognition train a support vector machine (SVM) for classification, an embedding review from paragraph vectors and product embedding from GRU were combined. According to [26], the efficacy of three distinct machine learning algorithms—Multinomial NB, linear SVM, and LSTM—was assessed through comparison, training, and testing using a dataset made up of product reviews from Amazon.com that could be classified binary as either positive or negative. The performance of LSTM for binary sentiment classification of Amazon.com product reviews was the most acceptable and was not significantly influenced by the product category from which the reviews were sourced.

A machine learning-based model was designed in [27] to spot discrepancies between customer reviews and the corresponding ratings on e-commerce websites. The consistency between reviews and ratings was investigated using an LSTM-based model that classified each review as either positive or negative polarity. The resulting polarity was compared with the customer rating to define its consistency. Three different ways were employed in the studies, anticipating how each review would rate on a scale of 1 to 5, categorizing reviews into three different categories—positive, negative, and neutral, and last, dividing reviews into two categories—positive and negative. The presented model was found to work well for two-class classification. Through opinion analysis of Amazon product reviews, Shah et al. [28] divided the reviews into favorable, neutral, and unfavorable product sentiments. Four machine learning techniques—logistic regression, multinomial NB, Bernoulli NB, and random forest—were employed to apply SA by classifying reviews after merging the data with some neutral and unfavorable sentiments and determining their accuracy. Noori [29] suggested a framework for classifying and forecasting consumer sentiments. an international hotel provided reviews from its customers. Six machine learning methods, including SVM, artificial neural network (ANN), Naïve Bayes (NB), decision tree (DT), C4.5, and k-nearest neighbor (K-NN), were employed to categorize sentiments from the customer evaluations.

In [30], a DL-based review rating prediction framework was presented; this framework has two stages according to model architectures of DL bidirectional GRU; the first stage was employed to predict polarity, and the second stage used the results from the first stage's polarity classes to anticipate review ratings from review text. Several tests were performed on the Amazon and Yelp datasets to assess the framework. LSTM and RNN-based models for sentiment classification and analysis were developed in [31]. Data collection, preprocessing, feature encoding, and classification were incorporated into the suggested models. The research employed various textual datasets, including Amazon Products, IMDB, cell phones and accessories, and Yelp datasets, to determine the significance of the proposed models. A semi-supervised short text sentiment classification approach based on an enhanced Bert model was presented in [32] for unlabeled and imbalanced quick text data sentiment analysis. The MixMatchNL approach, which integrates a sizably large number of unlabeled data with a relatively small quantity of labeled data to produce the labeled data, is used to create the improved data. The model's conventional cross-entropy loss function was upgraded to the Focal Loss function to address the data imbalance in short text datasets. The developed model, Text-CNN, LSTM, Bi-LSTM, and Bert models were assessed in the experiments on the public datasets for Amazon reviews and Chrome reviews.

3 The proposed methodology

This paper proposes a sentiment classification system based on DL for Amazon reviews. The proposed system consists of the following steps: preprocessing, converting the document to sequence, feature extraction, and classification. During text data analysis, data preprocessing is essential. The goal of text preprocessing is to convert unstructured text into a format that can be fed into models for further analysis and learning. Filtering data through preprocessing is a key aspect of data normalization. Among the steps in preprocessing data are normalization, word tokenization, eliminating stop words, and changing the text to lowercase. Data cleaning was achieved by implementing various tasks in this work.

3.1 Data Preprocessing

Sentiment analysis models do not take into account punctuation. Due to their lack of relevance to sentiment analysis, these punctuations need to be removed. So, erasing the Punctuation was the first preprocessing step. When customers share their reviews, they often write in a way that doesn't follow standard grammar rules. This means that their text may include both uppercase and lowercase letters, making it difficult for certain methods to classify their feedback correctly. The entire text was converted to a standardized format to address this issue. As the second preprocessing step, the "lower" function was used to convert all uppercase text to lowercase while keeping all other characters intact. So, we can help the classifier better determine the sentiment of the text. Finally, covert text to tokens. Tokenization is a powerful technique used to divide text streams into smaller, more manageable components. By breaking down larger pieces of content into fragments or phrases, tokenization helps to simplify the analysis of complex textual material. When tokens are applied, data mining becomes much more straightforward, making it an essential tool for lexical evaluation and highly beneficial in the fields of semantics and sentiment analysis.

3.2 The proposed DL method

CNNs learn to identify patterns across space, whereas RNNs are trained to recognize patterns over time [33]. RNNs succeed at NLP jobs requiring understanding long-range semantics, such as POS tagging or question answering. In contrast, CNNs succeed in situations requiring the recognition of local and position-invariant patterns. These patterns might represent important phrases that convey a sentiment. So, we proposed a SA and classification using CNNs.

The proposed CNNs contain an input layer, embedding layers, convolutional layers, pooling layers, a dropout Layer, a fully connected layer, and a classification layer. The convolutional and pooling layers encode the input tokens, while the other layers are considered prior encoding feature classifiers. Each convolutional filter extracts words or phrases from the raw text, creating an element for the encoded feature vector. The subsequent layers take the feature vector generated by the convolutional filters as input. Therefore, examining the CNN subsequence and classification layers could aid in determining the best convolutional filters.

Instead of working on serial and sequential layers, the proposed DL model implies the concept of feeding the input features into two different branches. Our deep learning employed two branches. This method typically comprises two branches, allowing the model to process input data through parallel routes. Each branch can specialize in capturing different input features, contributing to a more comprehensive representation.

The two-branch deep learning architecture employs parallel processing to enhance the model's ability to learn complex patterns and relationships in the data by simultaneously considering multiple perspectives or features. This method is used with deep learning to tackle the difficulties associated with training extremely deep neural networks. This method facilitates the flow of information across multiple layers more efficiently. This approach helps to address the vanishing gradient problem and makes it possible to train very deep networks.

The proposed deep learning model comprises two paths: the main and secondary paths. The main path consists of an input layer and an embedding layer. Meanwhile, the secondary path contains two branches that directly pass the input, or a slightly transformed version of it, from the embedding layer to a later layer in the network. The main innovation of this approach is that instead of having the main path learn to approximate the identity function, the network is explicitly forced to learn the difference between the input and the output. The following point can summarize the flow of features through the proposed deep learning model.

1 The input data are fed into two branches.
2 Each branch processes the input independently and learns different aspects of the underlying features.
3 The outputs from both branches are then joined together or concatenated.
4 The combined output is passed through an activation function, such as ReLU, and added to the original input data.

Using these complex architectures with less data can easily lead to over-fitting. Feeding the input into two branches can result in simpler models, while we have the benefit of complex structures. Using two branches instead of serial layers proves its ability to enhance performance and stability. This method overcomes dataset shortages, such as insufficient data, even if there is an imbalance between dataset labels. The proposed method feeds the output of the embedding layer to two different branches. In other meaning, the proposed model comprises two models in each branch. These branches consisted of convolutional, batch normalization, a nonlinear activation function, dropout, and global max pooling layer. Each convolutional layer consisted of 100 filters. The batch normalization layer was utilized to overcome the problem of changing the layer's parameter based on the previous one. Therefore, the normalization is done using batch normalization for each mini-batch during training.

A global max pooling layer is utilized to downscale the obtained feature matrices by setting the size of the pool equal to the input size of feature matrices. A fully connected layer converted the input size into an N-dimensional output vector. The global max pooling layer output from each branch is concatenated together.

overall description for different layers in the proposed DL method. Following clarifying the critical steps of the suggested DL method, the pseudo-code defining the proposed DL method is provided in Algorithm 1.

Algorithm 1 The proposed DL method

Input:
Amazon product reviews dataset
N – total Amazon reviews dataset records
$Epoch_{Max}$ – maximum number of allowed epochs ($Epoch_{Max} = 100$)
Output:
classify the customer reviews from 1 to 5 and obtain the predicted rating

Start
  Load Amazon reviews dataset;
  for record $i = 1 : N$ **do**
    Eliminate the extraneous words and special symbols that are not required, such as digits, stop-words, commas, hashtags, and punctuation marks;
    Convert all uppercase text words to lowercase to handle a unified form;
    Tokenize the resultant text words by segregating them into a bag of words (small tokens) to obtain the words that have value in the created matrix ;
  end for
  Update the hyperparameter by Adam optimizer;
  Embed the obtained tokens to the same lengths;
  for $epoch = 1 : Epoch_{Max}$ **do**
    Feed the tokens to the proposed DL model;
    Compute the Loss function for each epoch;
    if $loss(epoch) >= loss(epoch - 1)$ **then**
      Compute the Loss function for the next epoch ($epoch + 1$) and compare it with the previous epoch ($epoch$);
      if $loss(epoch + 1) >= loss(epoch)$ **then**
        Decrease the learning rate by 0.1;
      else
        go to 10
      end if
    end if
  end for
  Rate the customer review from 1 to 5;
  Compare the predicted customer review ratings resulting from the DL model with the actual customer review ratings from the dataset;
  if Predicted DL customer review ratings != actual customer review ratings **then**
    There is an inconsistency between the customer review and its actual rating;
  else
    There is consistency between the customer review and its actual rating;
  end if
End

fully connected layer: $\mathbb{R}_M \rightarrow \mathbb{R}_N$

$\mathbb{R}_M$ and $\mathbb{R}_N$ denote the input size and the number of classes, respectively. Finally, Softmax is an activation function for the final classification layer. This approach generates trustworthy and private feature maps from two sub-models, not just one, as with the CNNs model. So, the performance of the overall model is not significantly impacted by the mistake that happens in one branch. The design of the suggested architecture employs multiple paths with various hyperparameters for efficient feature extraction. The overall architecture is shown in Fig. 1. Table 1 gives the

The proposed DL model combines many features, such as increasing representation power, improving learning efficiency, and being flexible in branch design. So, combining features from two separate paths can result in more informative and richer representations of the input data. In addition, splitting the processing tasks into two branches allows for better learning and optimization than a single, complex branch. Finally, it is possible to customize each branch by adding different types and numbers of layers to capture specific aspects of the data. The next algorithm illustrates the overall process of the proposed method.

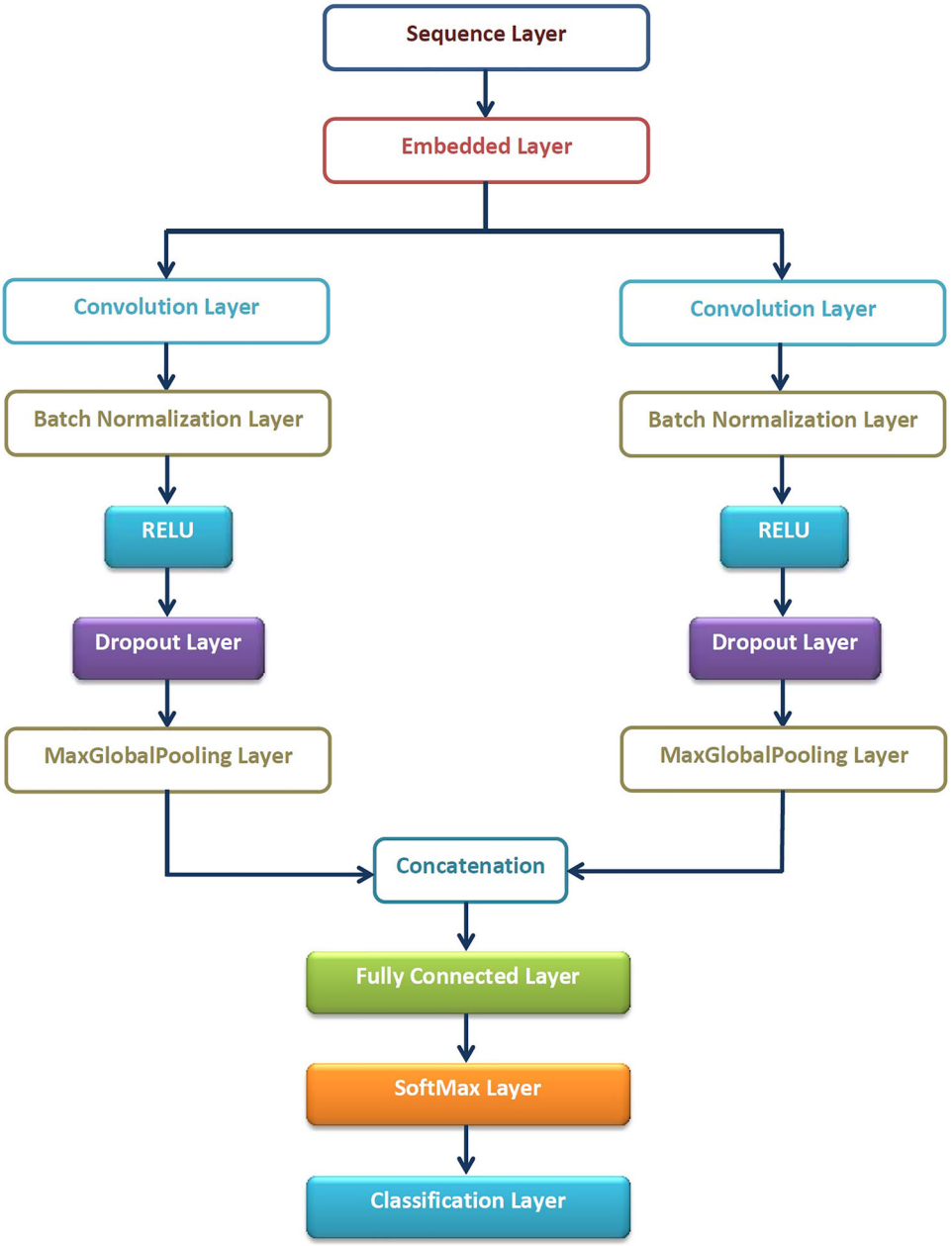

Fig. 1 Framework of the proposed methodology

4 Experimental results and analysis

This section shows the details of the experimental results to evaluate the proposed methodology, describes the evaluation measures, and discusses the classification results.

4.1 Dataset description

The dataset utilized in this study was extracted from Amazon product reviews [34], with a total of 3.5 million product reviews. Each review was recorded with its ID, the reviewer's name, the body, the date it was posted, the number of positive votes achieved, the review title, and the rating. In our case, however, we utilized only two attributes: the body of the review and the rating. The rating is based on a 5-star scale. In this study, we conducted three different experiments. In the first experiment, we utilized the proposed model in predicting five classes where each class represents a rating value (ratings 1 to 5). In the second experiment, we used the proposed model in predicting three categories: the negative class (class 0: ratings 1 and 2), the neutral class (class 1: rating 3), and the positive class (class 2: ratings 4 and 5). In the last experiment, we used the proposed model to predict the negative class (class 0: ratings 1 and 2) and positive class (class 1: ratings 3, 4, and 5). Figure 2 shows the data visualization of the dataset

Table 1 Full description of the proposed DL model

Layer type	Activations	Number of learnable
Sequence input	$1 \times 1 \times 1$	0
Word embedding layer	$100 \times 1 \times 1$	15141700
Convolution	$200 \times 1 \times 1$	40200
Batch normalization	$200 \times 1 \times 1$	400
ReLU	$200 \times 1 \times 1$	0
Dropout	$200 \times 1 \times 1$	0
1-D Global max pooling	200×1	0
Convolution	$200 \times 1 \times 1$	60200
Batch normalization	$200 \times 1 \times 1$	400
ReLU	$200 \times 1 \times 1$	0
Dropout	$200 \times 1 \times 1$	0
1-D Global Max Pooling	200×1	0
Concatenation	400×1	0
Fully connected	No of classes $\times$ 1	2005
Softmax	No of classes $\times$ 1	0
Classification output	No of classes $\times$ 1	0

used, representing the relationship between the number of reviews and ratings. According to Fig. 2, it is evident that reviews were highly unbalanced between different data classes.

4.2 Experiment setup

The proposed methodology was trained in an offline learning mode but using online resources on the Kaggle dataset [34], and the hyperparameters were tuned using Adam. The hyperparameters in our architecture are:

- Mini Batch Size=128.
- Epochs = 100.
- Initial Learn Rate=0.001.
- Learn Rate Schedule='piecewise'.
- Validation Patience = inf.
- Learn Rate Drop Factor=0.1.
- Learn Rate Drop Period=2.
- Optimizer = Adam.

Parameters are randomly initialized and updated using Adam, while scheduling annealing was used to drop the learning rate every two epochs. Scheduling annealing was used to accelerate the optimizer convergence and achieve minimum errors. Scheduled annealing is proposed with Adam to update the network parameters. This algorithm helps avoid local minima and saddle points, and convergence to the global optimum solution is made possible by the scheduled annealing [35]. The dataset is split into learning and testing. To run all experiments in this study, MATLAB 2022a 64 bits was used on a computing environment with a Dual Intel® Xeon® Gold 5115 2.4 GHz CPU and 128 GB of RAM on the operating system Microsoft Windows Server 2019. Thus, 80% of the data was used for learning, while 20% was used for evaluating the proposed approach. Finally, a tenfold cross-validation method is employed to reduce model error for learning and testing purposes.

4.3 Evaluation measures

In this paper, to ensure that the experimental results are statistically reliable, the efficiency of the proposed methodology must be evaluated using standard measures. There are specific terms that need to be defined before talking about these standard metrics, as follows:

- **True Positive** (T_P): represents the percentage of truthful reviews that are successfully categorized using the proposed methodology.
- **True Negative** (T_N): defines the percentage of false reviews that are successfully categorized using the proposed methodology.
- **False Positive** (F_P): represents the percentage of truthful reviews categorized as false reviews.
- **False Negative** (F_N): describes the percentage of false reviews categorized as truthful reviews.

To that end, the main assessment measures utilized in this paper are accuracy, precision, recall, $F1$-score, sensitivity, and specificity. Accuracy determines how frequently the classifier makes the right prediction, which is the number of successful predictions $(T_P + T_N)$ divided by the total number of predictions $(T_P + T_N + F_P + F_N)$. Precision gauges the quality of a classifier, which is the True Positive prediction T_P divided by the total number of positive classified predictions $(T_P + F_P)$. Recall determines how many positive data it returns, which is the True Positive prediction T_P divided by the sum of True Positive and False Negative predictions $(T_P + F_N)$. $F1$-score estimates the accuracy of a classifier, which incorporates both precision and recall measures by calculating their harmonic average. Sensitivity specifies the number of positive data rightly predicted, which is similar to recall. Specificity defines the number of negative data rightly predicted, which is the True Negative prediction T_N divided by the sum of True Negative and False Positive predictions $(T_N + F_P)$.

4.4 Results analysis

In this part, we compared the empirical outcomes of the proposed methodology with other methods on the Amazon customer reviews dataset. The suggested system and selected methods are executed on a framework with

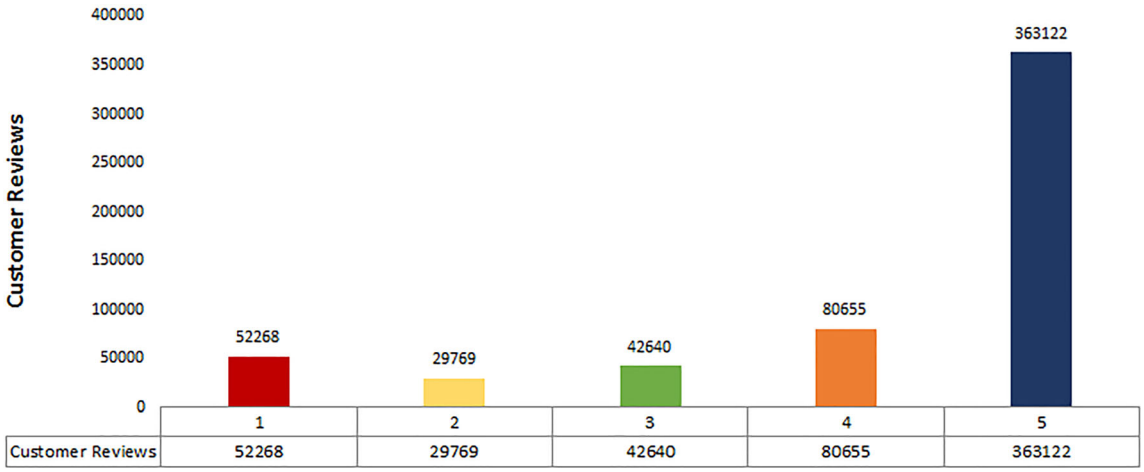

Fig. 2 Data visualization of the number of reviews versus ratings

identical parameters and tested on the Amazon customer reviews dataset for a reliable comparison. The results in Table 2 reflect the performance of the proposed method for predicting five classes of product reviews in terms of accuracy, specificity, precision, recall, and the $F1$-score, where boldface numbers indicate the best results. The proposed methodology was compared with four counterparts (LSTM [27], Bi-LSTM, Text-CNN, and Bert (Bidirectional Encoder Representations from Transformers) [32]). The proposed methodology ranked first in classification accuracy, while Bi-LSTM ranked first in the F1-score. According to the utilized sample of Amazon reviews, the proposed method used the most significant sample size of 568,454 records. Figure 3 compares the proposed methodology and peers' methods regarding the accuracy and $F1$-score. According to Fig. 3, the proposed achieved the highest classification accuracy, but Bi-LSTM achieved the highest results in terms of the $F1$-score. Figure 4 displays the Confusion matrix of the proposed methodology for predicting five classes. Finally, the sensitivity and specificity curves of the proposed method for predicting five categories are shown in Fig. 5.

The performance of the proposed method for predicting three product reviews in terms of accuracy, specificity, precision, recall, and the $F1$-score is displayed in Table 3, in which boldface numbers indicate the best results. The proposed methodology was compared with eight counterparts (LSTM [27], paragraph vectors, paragraph vectors with GRU [25], logistic regression, multinomial NB, Bernoulli NB [28], weighted uni-grams-SVM, and weighted Unigrams-NB [24]). According to the results in Table 3, the proposed methodology ranked first in all performance measures. Logistic regression ranked second in all evaluation metrics with a dataset of 35,000 product reviews. The paragraph vectors approach and paragraph vectors with GRU obtained the worst results but utilized the most extensive product reviews compared with other counterparts. Figure 6 compares the proposed methodology and peers' methods regarding the $F1$-score. According to Fig. 6, the proposed method achieved the highest $F1$-score, but multinomial NB achieved the worst results in terms of the $F1$-score. Figure 7 displays the confusion matrix of the proposed methodology for predicting three classes. Finally, the sensitivity and specificity curves of the proposed method for predicting three categories are shown in Fig. 8.

The results in Table 4 show the proposed method's performance for predicting two product reviews in terms of accuracy, specificity, precision, recall, and the $F1$-score. Note that boldface values denotes the best results. The proposed methodology was compared with five counterparts (LSTM [27], linear SVM, multinomial NB [26], weakly supervised deep embedding (WDE)-CNN, and WDE-LSTM [22]). According to the results displayed in Table 4, The proposed methodology ranked first in all performance measures except precision. Multinomial NB ranked first in terms of precision. Linear SVM obtained the worst results in all performance measures. According to the utilized sample of Amazon reviews, the proposed method used the most significant sample size of 568,454 records. Figure 9 compares the proposed methodology and peers' methods regarding the accuracy and $F1$-score. According to Fig. 3, the proposed achieved the highest classification and $F1$-score. Figure 10 displays the Confusion matrix of the proposed methodology for predicting two classes. Finally, the sensitivity and specificity curves of the proposed method for predicting two categories are shown in Fig. 11.

4.5 Consistency examination

After classifying the customer reviews and obtaining their predicted rating based on the proposed DL model, the final

Table 2 Results of the proposed methodology for predicting 5 classes

Method	Sample Size	class	Accuracy	Specificity	Precision	Recall	*F*1-score
Proposed Methodology	568,454	1	0.95	0.97	0.77	0.76	0.76
		2	0.95	0.97	0.51	0.57	0.54
		3	0.94	0.96	0.56	0.62	0.59
		4	0.89	0.92	0.52	0.65	0.58
		5	0.88	0.84	0.94	0.88	0.91
		Average	**0.92**	**0.93**	**0.66**	**0.69**	0.67
LSTM [27]	29,163	1	–	–	0.63	0.77	0.70
		2	–	–	0.33	0.00	0.01
		3	–	–	0.34	0.15	0.21
		4	–	–	0.41	0.23	0.30
		5	–	–	0.67	0.88	0.76
		Average	–	–	0.47	0.41	0.40
Text-CNN [32]	72,500	Average	0.85	–	–	–	0.85
Bi-LSTM [32]	72,500	Average	0.90	–	–	–	0.90
Bert [32]	72,500	Average	0.91	–	–	–	0.89

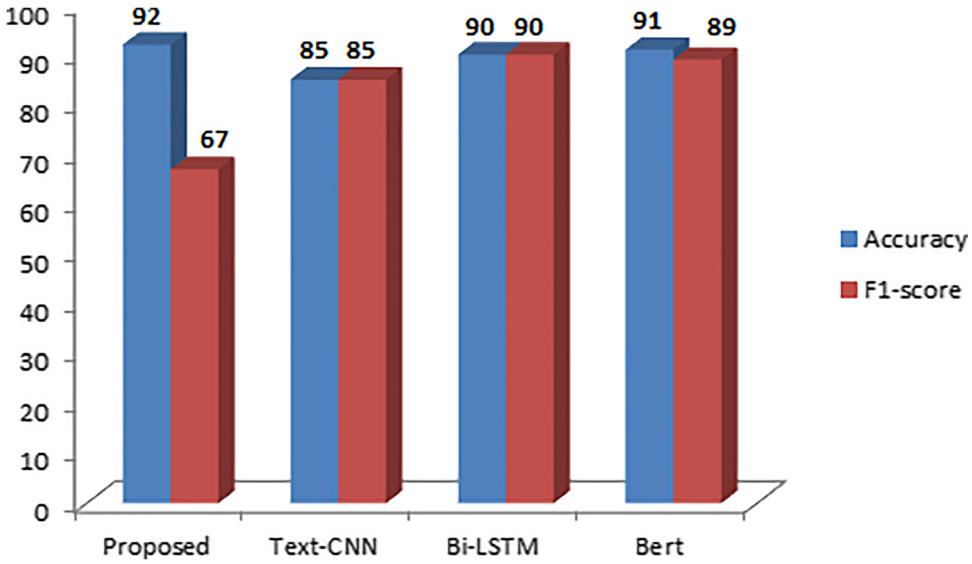

Fig. 3 Classification accuracy and *F*1-score of the proposed methodology and counterparts methods for predicting five classes

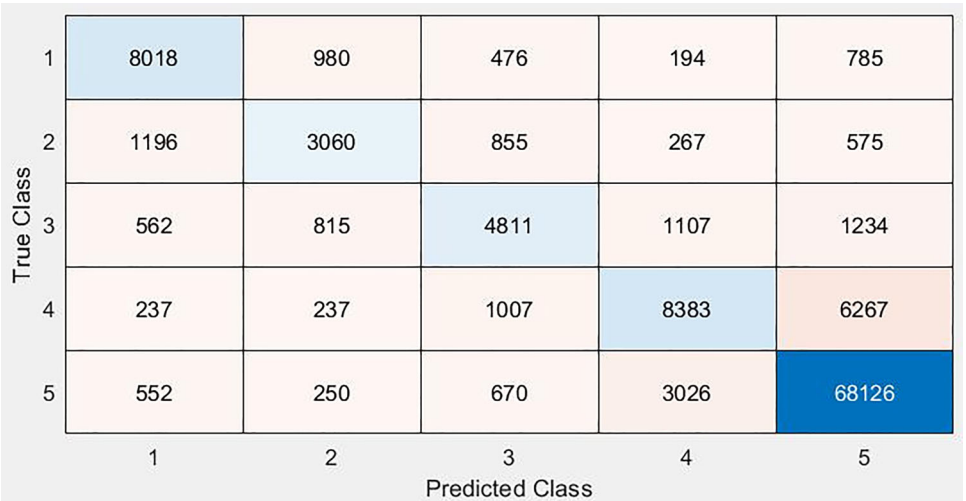

Fig. 4 Confusion matrix of the proposed methodology for predicting 5 classes

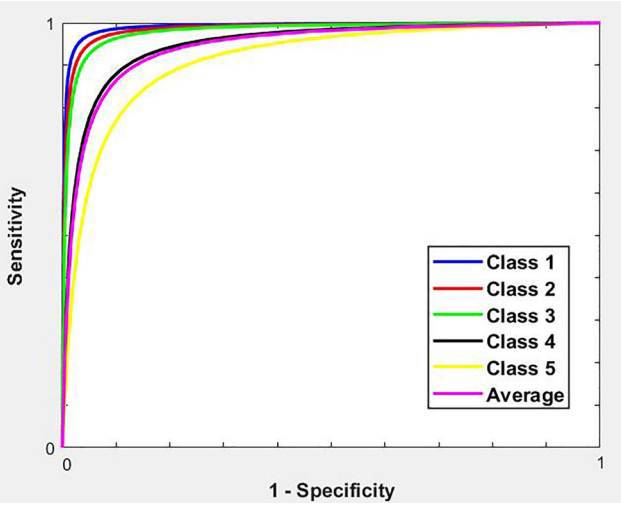

Fig. 5 Sensitivity and specificity curves of the proposed methodology for predicting five classes

step remains, which involves comparing the predicted customer review ratings resulting from the proposed DL model and the actual customer review ratings to clarify whether there is consistency between them or whether the customer review rating is inconsistent. In case the proposed DL model classifies the customer review according to the classification of review polarity as a negative rating and the actual customer rating is a 1 or 2, then we can conclude that there is consistency between the customer review and its actual rating; otherwise, for customer ratings 3, 4, or 5 we can infer that there is inconsistency between the customer review and its actual rating. On the other hand, if the proposed DL model classifies the customer review as a positive rating and the actual customer rating is 1 or 2, then we can deduce that there is an inconsistency between the customer review and its actual rating; otherwise, for customer ratings 3, 4, or 5 we can infer that there is consistency between the customer review and its actual rating.

4.6 Managerial implications and advantages

The system we propose provides numerous managerial implications and advantages. To begin with, it can be seamlessly integrated into an already established online review system. Additionally, it can be utilized as a consistency checker before publishing any review rating on the product webpage. Lastly, the system can automatically generate a rating on a 1-5 scale, and reviewers can be prompted to write a review based on sentiments.

Table 3 Results of the proposed methodology for predicting three classes

Method	Sample Size	class	Accuracy	Specificity	Precision	Recall	*F*1-score
Proposed Methodology	568,454	0	0.95	0.97	0.83	0.83	0.83
		1	0.94	0.96	0.54	0.64	0.58
		2	0.94	0.87	0.97	0.95	0.96
		Average	**0.95**	**0.93**	**0.78**	**0.81**	**0.79**
LSTM [27]	29,163	0	–	–	0.70	0.75	0.72
		1	–	–	0.89	0.97	0.93
		2	–	–	0.71	0.01	0.02
		Average	–	–	0.76	0.58	0.56
Paragraph vectors [25]	3.5 M	Average	0.81	–	0.59	0.41	–
Paragraph vectors and GRU [25]	3.5 M	Average	0.82	–	0.59	0.43	–
Logistic regression [28]	35,000	0	–	–	0.79	0.63	0.70
		1	–	–	0.62	0.35	0.45
		2	–	–	0.93	0.98	0.96
		Average	–	–	**0.78**	0.65	0.70
Multinomial NB [28]	35,000	0	–	–	0.81	0.25	0.38
		1	–	–	0.54	0.01	0.02
		2	–	–	0.87	1.00	0.93
		Average	–	–	0.74	0.42	0.44
Bernoulli NB [28]	35,000	0	–	–	0.61	0.48	0.54
		1	–	–	0.37	0.23	0.28
		2	–	–	0.91	0.95	0.93
		Average	–	–	0.63	0.55	0.58
Weighted Uni-grams-SVM [24]	24,500	Average	0.81	–	–	–	–
Weighted Uni-grams-NB [24]	24,500	Average	0.77	–	–	–	–

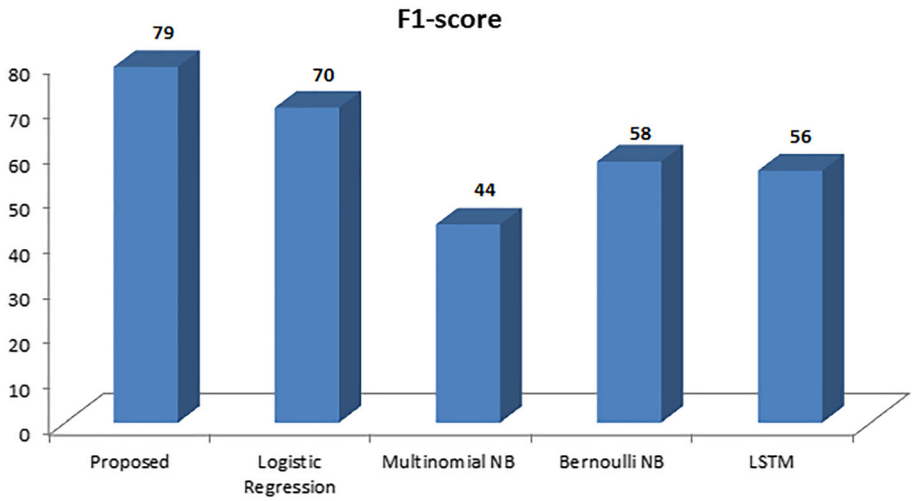

Fig. 6 *F*1-score of the proposed methodology and counterparts methods for predicting three classes

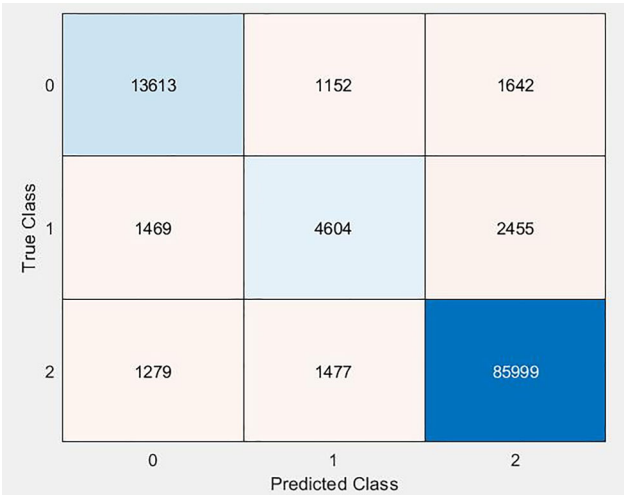

Fig. 7 Confusion matrix of the proposed methodology for predicting three classes

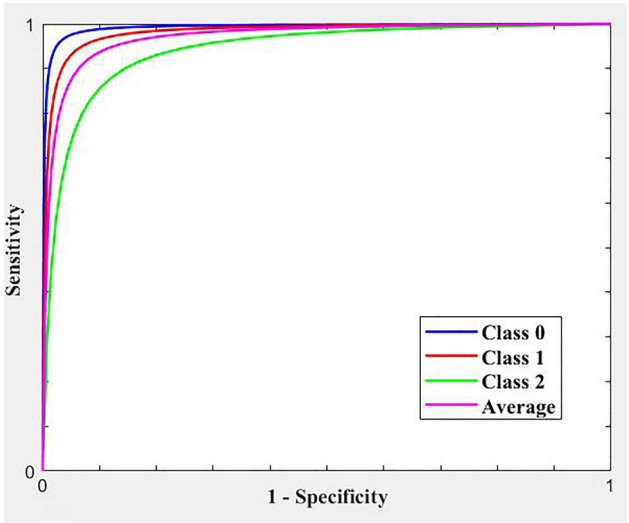

Fig. 8 Sensitivity and specificity curves of the proposed methodology for predicting three classes

Nonetheless, the proposed system must be learned in this scenario to predict ratings within the 1–5 range, as demonstrated in 2.

5 Conclusions and future directions

Most customers on e-commerce websites rely on reviews and accompanying ratings to evaluate the quality of products. There is frequently an inconsistency between reviews and ratings, making it challenging to identify pertinent information amid many reviews. This paper proposed an effective DL model based on the flow of the feature into different branches instead of working on serial and sequential layers for classifying customer reviews and checking for inconsistency between reviews and ratings. The performance and stability of the suggested DL model were evaluated on a sizable dataset. We conducted three separate experiments for this study. In the first experiment, the positive class (class 1: ratings 1 and 2) and negative class (class 2: ratings 3, 4, and 5) were predicted using the suggested model. Secondly, We employed the proposed model to expect three classes: positive class (class 1: ratings 1 and 2), neutral class (class 2: rating 3), and negative class (class 3: ratings 4 and 5). Finally, we employed the suggested model to anticipate five classes, each corresponding to a rating value (ratings 1, 2, 3, 4, and 5). The experimental outcomes demonstrated that the performance of the proposed model is much better than its counterparts in terms of prediction accuracy and other performance criteria.

However, this paper has some limitations that need to be investigated. Firstly, Amazon doesn't ask clients to add the real date of the visit, which may introduce biased outcomes. Additionally, the suggested DL model robustness

Table 4 Results of the proposed methodology for predicting two classes

Method	Sample size	class	Accuracy	Specificity	Precision	Recall	*F*1-score
Proposed methodology	568,454	0	0.95	0.96	0.79	0.87	0.83
		1	0.95	0.87	0.98	0.96	0.97
		Average	**0.95**	**0.92**	0.89	**0.92**	**0.90**
LSTM [27]	29,163	0	–	–	0.76	0.85	0.80
		1	–	–	0.97	0.95	0.96
		Average	—	—	0.86	0.90	0.88
Linear SVM [26]	60,000	Average	0.86	–	0.86	0.86	0.86
Multinomial NB [26]	60,000	Average	0.90	–	**0.92**	0.87	**0.90**
WDE-CNN [22]	11,754	Average	0.88	–	–	–	0.88
WDE-LSTM [22]	11,754	Average	0.88	–	–	–	0.88

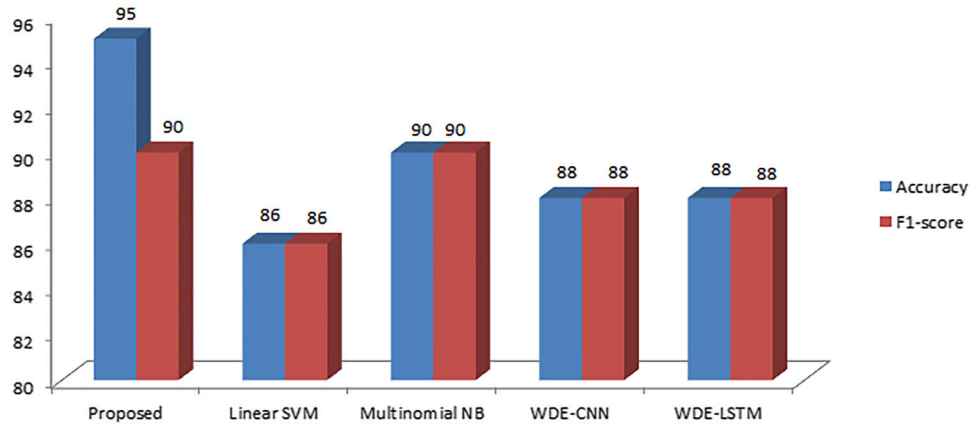

Fig. 9 Classification accuracy and *F*1-score of the proposed methodology and counterparts methods for predicting two classes

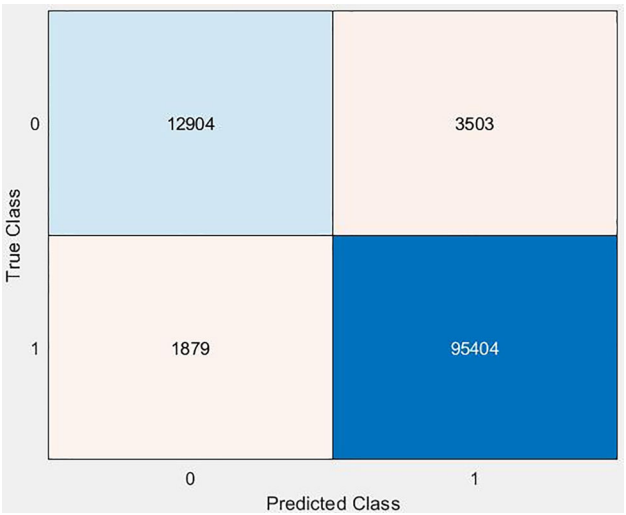

Fig. 10 Confusion matrix of the proposed methodology for predicting two classes

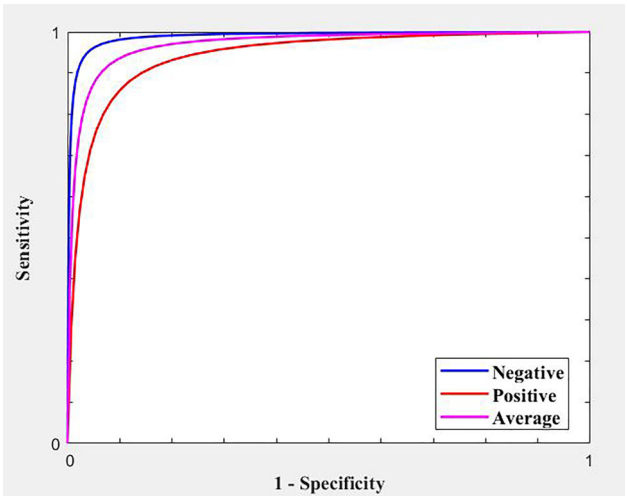

Fig. 11 Sensitivity and specificity curves of the proposed methodology for predicting two classes

requires to be validated via additional review platforms. Also, the proposed DL model encounters difficulties in accurately addressing nuanced language issues like sarcasm, irony, and subjective reviews that stem from the ambiguity of the language used and the speaker or writer's intention, which is characterized by the deliberate use of words and may contradict the literal meaning. Finally, although the DL model supports experimenters with an efficient route to create inferences about complex data with large volumes, it is often criticized for being a black box model with untraceable and unknown forecasts.

In future, an in-depth investigation is required to experiment with more platforms. Improving the preprocessing stages using methods such as integrating weighted word embeddings via TF-IDF is a viable solution to increase the classification accuracy further. Other machine learning-based models, especially unlabeled learning techniques, could be employed for further analysis. The capacity of the proposed DL model in other languages, such as Arabic, must also be explored. In addition, the inclusion of various machine learning-based models with different meta-heuristic optimization algorithms could be discussed. A sarcasm detector is needed to accurately explore sarcastic phrases in customer reviews by analyzing their characteristics compared to non-sarcastic ones, which improves the understanding of subtle communication patterns in online interactions. Last but not least, this study will be continued and expanded so that the suggested DL model can be applied to a variety of reviews and datasets, where the suggested DL model developed using Amazon customer reviews as training data may also be used with many other datasets that lack numerical rating values. For example, it can set ratings to comments on YouTube or Twitter or just forecast the rating of comments from different online stores that only allow text reviews with no rating option. Thus, this model can enhance the customer experience. This effort will also increase the capability of the proposed model so that it can be utilized in domains other than e-commerce, such as fake news detection, natural language processing, visual recognition, and fraud detection.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

Data availability Data are available on request from the authors.

Declarations

Conflict of interest All the authors declare that they have no conflict of interest.

References

1. Singh JP, Irani S, Rana NP, Dwivedi YK, Saumya S, Roy PK (2017) Predicting the"helpfulnes"of online consumer reviews. J Bus Res 70:346–355
2. Abd El-Mageed A, A, Abohany A. A., Elashry A, (2023) Effective feature selection strategy for supervised classification based on an improved binary aquila optimization algorithm. Comput Ind Eng 181:109300
3. Guo J, Wang X, Wu Y (2020) Positive emotion bias: role of emotional content from online customer reviews in purchase decisions. J Retail Consum Serv 52:101891
4. Bhuvaneshwari P, Rao AN, Robinson YH, Thippeswamy M (2022) Sentiment analysis for user reviews using bi-lstm self-attention based CNN model. Multim Tools Appl 81(9):12405–12419
5. Eslami SP, Ghasemaghaei M (2018) Effects of online review positiveness and review score inconsistency on sales: a comparison by product involvement. J Retail Consum Serv 45:74–80
6. Abd El-Mageed A, A, Gad AG, Sallam KM, Munasinghe K, Abohany AA, (2022) Improved binary adaptive wind driven optimization algorithm-based dimensionality reduction for supervised classification. Comput Ind Eng 167:107904
7. Palahan S (2023) Comparative analysis of deep learning models for predicting online review helpfulness, in Proceedings of the 2023 Asia Conference on Computer Vision, Image Processing and Pattern Recognition, pp. 1–5
8. Salminen J, Kandpal C, Kamel AM, Jung S-G, Jansen BJ (2022) Creating and detecting fake reviews of online products. J Retail Consum Serv 64:102771
9. Ashraf S, Rehman F, Sharif H, Kim H, Arshad H, Manzoor H (2023) Fake reviews classification using deep learning, In : International Multi-disciplinary Conference in Emerging Research Trends (IMCERT), vol. 1. IEEE 2023:1–8
10. Derbentsev VD, Bezkorovainyi VS, Matviychuk AV, Pomazun OM, Hrabariev AV, Hostryk AM (2023) A comparative study of deep learning models for sentiment analysis of social media texts. In CEUR Workshop Proceedings, pp. 168–188
11. Park EL, Cho S, Kang P (2019) Supervised paragraph vector: distributed representations of words, documents and class labels. IEEE Access 7:29051–29064
12. Socher R, Perelygin A, Wu J, Chuang J, Manning C. D, Ng A. Y, Potts C (2013) Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 conference on empirical methods in natural language processing, 1631–1642
13. Li Z, Liu F, Yang W, Peng S, Zhou J (2021) A survey of convolutional neural networks: analysis, applications, and prospects, IEEE Trans Neural Netw Learn Syst
14. Kalchbrenner N, Grefenstette E, Blunsom P (2014) A convolutional neural network for modelling sentences, arXiv preprint arXiv:1404.2188
15. Tang D, Qin B, Liu T (2015) Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 conference on empirical methods in natural language processing, 1422–1432
16. Sherstinsky A (2020) Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. Phys D Nonlin Phenom 404:132306
17. Van Houdt G, Mosquera C, Nápoles G (2020) A review on the long short-term memory model. Artif Intell Rev 53:5929–5955
18. Tang D, Qin B, Liu T, Yang Y (2015) User modeling with neural network for review rating prediction. In: Twenty-fourth international joint conference on artificial intelligence,

19. Chen T, Xu R, He Y, Xia Y, Wang X (2016) Learning user and product distributed representations using a sequence model for sentiment analysis. IEEE Comput Intell Magaz 11(3):34–44
20. Sharma S, Sharma S, Athaiya A (2017) Activation functions in neural networks. Towards Data Sci 6(12):310–316
21. Seo S, Huang J, Yang H, Liu Y (2017) Representation learning of users and items for review rating prediction using attention-based convolutional neural network. In: International Workshop on Machine Learning Methods for Recommender Systems
22. Zhao W, Guan Z, Chen L, He X, Cai D, Wang B, Wang Q (2017) Weakly-supervised deep embedding for product review sentiment analysis. IEEE Trans Knowl Data Eng 30(1):185–197
23. Martín CA, Torres JM, Aguilar RM, Diaz S et al. (2018) Using deep learning to predict sentiments: case study in tourism, Complexity
24. Rathor AS, Agarwal A, Dimri P (2018) Comparative study of machine learning approaches for amazon reviews. Procedia Comput Sci 132:1552–1561
25. Shrestha N, Nasoz F (2019) Deep learning sentiment analysis of amazon.com reviews and ratings. Int J Soft Comput Artif Intell Appl 8(1):01–15, doi.org/10.5121%2Fijscai.2019.8101
26. Güner L, Coyne E, Smit J (2019) Sentiment analysis for amazon.com reviews, Big Data in Media Technology (DM2583) KTH Royal Institute of Technology, Stockholm,
27. Saumya S, Singh J. P, Kumar A (2021) A machine learning model for review rating inconsistency in e-commerce websites, in Data Management, Analytics and Innovation: Proceedings of ICDMAI 2020, Volume 1.Springer, 221–230
28. Shah BK, Jaiswal AK, Shroff A, Dixit AK, Kushwaha ON, Shah NK (2021) Sentiments detection for amazon product review. In: International Conference on Computer Communication and Informatics (ICCCI) 2021:1–6
29. Noori B (2021) Classification of customer reviews using machine learning algorithms. Appl Artif Intell 35(8):567–588
30. Ahmed BH, Ghabayen AS (2022) Review rating prediction framework using deep learning. J Amb Intell Humaniz Comput 13(7):3423–3432
31. Iqbal A, Amin R, Iqbal J, Alroobaea R, Binmahfoudh A, Hussain M (2022) Sentiment analysis of consumer reviews using deep learning. Sustainability 14(17):10844
32. Zou H, Wang Z (2023) A semi-supervised short text sentiment classification method based on improved bert model from unlabelled data. J Big Data 10(1):1–19
33. Lin C-J, Jeng S-Y, Chen M-K (2020) Using 2d CNN with Taguchi parametric optimization for lung cancer recognition from ct images. Appl Sci 10(7):2591
34. King E, Amazon customer reviews, 2016. [Online]. Available: https://www.kaggle.com/datasets/vivekprajapati2048/amazon-customer-reviews?datasetId=1470538
35. Naguib SM, Hamza HM, Hosny KM, Saleh MK, Kassem MA (2023) Classification of cervical spine fracture and dislocation using refined pre-trained deep model and saliency map, Diagnostics, 13(7)