Multimedia Tools and Applications
https://doi.org/10.1007/s11042-021-10602-y

Spam review detection using self attention based CNN and bi-directional LSTM

P. Bhuvaneshwari[1] · A. Nagaraja Rao[1] · Y. Harold Robinson[2]

Received: 18 August 2020 / Revised: 23 November 2020 / Accepted: 25 January 2021
Published online: 15 February 2021

Abstract

Opinion reviews are a valuable source of information in e-commerce. Indeed, it benefits users in buying decisions and businesses to enhance their quality. However, various greedy organizations employ spammers to post biased spam reviews to gain an advantage or to degrade the reputation of a competitor. This results in the explosive growth of opinion spamming. Due to its nature and their increasing volume, spam reviews are a fast-growing serious issue on the internet. Until now, researchers have developed many Machine Learning (ML) based methods to identify opinion spam reviews. However, the traditional ML methods cannot effectively detect spam messages due to the limited feature representations and the data manipulations done by spammers to escape from the detection mechanism. As an alternative to ML-based detection, in this paper, we proposed a Deep Learning (DL) based novel framework called Self Attention-based CNN Bi-LSTM (ACB) model to learn document level representation for identifying the spam reviews. Our approach computes the weightage of each word present in the sentence and identifies the spamming clues exists in the document with an attention mechanism. Then the model learns sentence representation by using Convolution Neural Network (CNN) and extracts the higher-level n-gram features. Then finally, sentence vectors are combined using Bi-directional LSTM (Bi-LSTM) as document feature vectors and identify the spam reviews with contextual information. The evaluated experiment results are compared with its variants and the result shows that ACB outperforms other variants in terms of classification accuracy.

Keywords E-commerce · Opinion spam reviews · Machine learning · Deep learning · Self attention-based CNN Bi-LSTM (ACB) model · Convolution neural network · Self-attention mechanism · Bidirectional long short term memory

✉ Y. Harold Robinson
yhrobinphd@gmail.com

Extended author information available on the last page of the article

1 Introduction

The internet has developed dramatically over the past two decades, resulting in the establishment of numerous tools to empower people. E-commerce is a business model where selling or buying products and services is carried out electronically. There are many advantages of using e-commerce like the availability of complete information of a given product including several payment options, access to purchase a given product at anytime and anywhere, instant access to new arrivals, and personalized offers. Through the accessibility granted by modern web technologies, any user can share their feedback in the form of reviews or ratings in an e-commerce application. This information is useful to the business unit to identify if there are any issues related to their products that need further improvement in terms of Research & Development. This assists business units to come up with better products that are catchy from the customer's point of view, and in turn leads to huge revenue. These opinion reviews are also helpful to the active customer who read them to figure out the other customer's experience about that product. This allows the active user to take decision making as either purchase that product or not.

According to the survey conducted by Horrigan [9], 81% of online buyers have analyzed the product with the information available on the Internet before they have a plan to buy and 79% of them are feeling confident in making the right decision. Seen as an opportunity, many companies hire people to post spam reviews to promote their products or to relegate the products of a competitor. Since there is no control mechanism available, anyone can write anything on the e-commerce websites and a huge amount of deceptive reviews is easily generated. These reviews are fictitious which are deliberately scripted to sound authentic [7]. These reviews are generally short which are created anonymously by fake reviewers with various intentions. Several kinds of spam are generated especially on social media, blogs, email, web forums, and SMS. The first investigation on spam review has been done in the web page and e-mail domains [3]. Due to the rapid growth of e-commerce applications, recent research on customer review spam detection plays a major area of research. Spam opinion reviews are more deceitful than the above-cited spams. Ye and Akoglu [27] have been reported that more than 33% of the reviews on the Internet are spam reviews and are increasing. Recently, Luca and Zervas [16] identified that 16% of Yelp restaurant reviews are deceptive. In the test of Ott et al. [19], it is proved that only 57.33% of average accuracy is achieved from three human judges in identifying spam reviews. There is an increasing number of users who are worrying about taking the wrong decisions by these fake reviews in online shopping [10]. Consequently, more efforts are required to identify misleading fake opinion reviews to ensure the Web's credibility. This makes spam review detection a hot topic in the research field. In this study, we have used the terms "opinion spam," fake reviews," "deceptive reviews", and "spam reviews" interchangeably.

Generally various researches have been done in identifying spam detection from three different angles: detecting opinion spam, detecting spammers, and detecting the networks of spammers. To detect the spamming activities, features such as review-content specific features, reviewer-behavior specific features, and review-reviewer network specific features are utilized effectively. Researchers believed that spammers leave some clues in review writing, so they have developed so many techniques such as linguistic and psychotic features, n-gram methods, POS tagging, and duplicity measures to identify spam reviews and to retain the factual customers [8]. By evaluating these models Dong et al. [2] stated that these models considered only explicit information and suggested that mining the implicit information is the key for spam detection. Ren and Ji [22] stated that due to sparsity it is difficult to obtain semantic information by using linguistic features. Feng et al. [6] investigated syntactic stylometry to identify spam reviews and identified that the writing style of spammers differs from the genuine reviewer. The majority of the existing approaches follows Ott et al. [19] and utilizes machine learning algorithms. Most of the researchers employed supervised learning to train the classifier and to extract the features. But due to the increasing number of reviews and review sites, traditional ML algorithms could not guarantee the accuracy in spam identification with limited feature representation [1]. It is hard for ML to learn the inherent law of data from a semantic perspective. Also due to feature sparsity, it's difficult for the ML algorithms to capture the non-local semantic information over a sentence and represent it in a document with the viewpoint of global discourse structures. In recent years, deep learning methods are adopted in various applications due to the automatic capability of feature extraction. It learns the hierarchical representations through several processing layers. It achieves highly competitive results in the context of Natural Language Processing (NLP). These models can learn global semantic representations by combining the features automatically using real-valued hidden layers. Zhang et al. [29] proposed a new approach using a recurrent convolution neural network known as DRI-RCNN. They used RCNN to capture the contextual local information of each word and emphasized that the contextual information plays a major role in classifying the review as fake or non-fake. But most of the existing work in the deep neural network fails to consider the contextual semantic information, implicit information, and document representation while detecting the deceptive reviews. To addresses the above limitations, we propose a novel deep neural network framework called ACB to classify spam reviews more accurately. The objective of this paper is to classify online reviews into fake or non-fake by considering the implicit syntactic contextual information in the reviews. First, neural networks take word embedding vectors as input from raw text datasets. Second, self-attention mechanism is used to focus on each word and strengthen the distribution of weights to the variable-length sequences. Third, a convolution neural network is used to reduce the dimensions of data by extracting the n-gram features and represent the sentence. Finally, Bi-LSTM is used to extract the contextual information and construct document representations from the sentence vectors outputted by the convolutional layer. Finally, a dense layer uses the document representation to categorize the review as fake or non-fake. For verifying the efficiency of the proposed model, the performance is measured with metrics, and the model is compared with its variants.

The rest of this paper is structured as follows. In section 2, we discuss the various works done in the field of fake review detection. Section 3 presents a detailed description of the proposed methodology. In section 4, the experimentation and results are discussed. Finally, Section 5 presents a summary of the work and future work.

2 Related work

The issue in review spam detection was first addressed by Jindal and Liu [10]. They used a product review dataset from Amazon. Though there is a huge volume of reviews available online, the major problem in the field of review spam detection is that the collection of enough real-world balanced class datasets and accurately labeling them to train the classifier that are classified the deceptive reviews into three categories such as untruthful reviews, reviews on brands, and non-reviews. Untruthful reviews give a positive impression to promote their products and damage the reputation of competitors by expressing a negative impression towards their products. Reviews on brands are mainly associated with manufacturers of the product, whereas non-reviews deal with advertisements and random text that has no opinions. By considering this work as a base, so many researchers have tried to explore the features of opinion spam to solve the problem effectively. They assumed that the writing style of the deceptive reviewers differs from the genuine reviewers. Yoo and Gretzel [28] manually compared the linguistic variations between them by comparing the 40 truthful and 42 deceptive reviews. Various research works have been done to identify the behavioral features of the spammers [18, 21].

In general, individual spammers duplicate the same review on different products or post the same review by using different user ids as if the reviews were posted by different reviewers. It is also possible that a group of spammers who work together and each has a unique user id through which they post nearly the same review on a given product. The primary issue in labeling the untruthful reviews is due to a lack of clear demarcation of words to classify them as fake or non-fake.

Recently, detection of opinion spam reviews by using Neural Networks proved to outperform conventional ML techniques [22, 23]. But very few models have been adopted deep learning and neural network architectures for identifying spam reviews. Wang and Chen [26] proposed a model using Long Short Term Memory [LSTM] and evaluated the performance by comparing it with ML algorithms. They proved that the LSTM performs better than the traditional ML methods. Document-level opinion spam review detection using a neural network has been performed by Ren and Ji [22]. They represented words with the corresponding continuous feature vectors from the look-up table. In general, either CNN or Recurrent Neural Network (RNN) can be used to model sentence-level or document-level modeling. However, they found that the performance can be improved by modeling sentences with CNN and document modeling with RNN. By giving word vectors as input to the Convolutional layer, local semantics were captured and the output from Convolutional filters is averaged out to capture the semantics of the whole sentence in the form of a sentence vector. By giving a set of sentence vectors as input to a Gated Recurrent Neural Network (GRNN), they obtained the document vector which represents the whole document. They concluded that by adding discrete features along with automatic neural features, the performance of the network can be improved as compared to discrete feature-based machine learning models. They suggested that the performance of the network can be increased further by applying attention mechanisms over sentence vectors [23]. Sedighi et al. [24] used Bi-LSTM along with a multi-headed self-attention mechanism to detect opinion spam on a cross-domain dataset like the one used by Ren and Ji [22]. Instead of using a convolutional layer like the other works [12, 22], they fed the word embeddings directly to a Bi-LSTM layer to capture the long-distance relation among the words. On top of the Bi-LSTM layer, they used a multi-headed self-attention mechanism and found that attention mechanism added value to the performance.

Zhao et al. [32] experimented the CNN by embedding the word order characteristics in its convolution layer and pooling layer and emphasized CNN is more suitable for identifying the short deceptive reviews. Li et al. [13] proposed a Sentence Weighted

Neural Network based on Sentence CNN to learn the weights of each sentence and represent in the document level. Wang et al. [25] proposed an attention-based neural network which used linguistic and behavioral features to detect the spam review. The behavioral features are obtained by using Multilayer Perceptron and linguistic features are obtained by using CNN. Top of that layer, the attention module is built to detect the spam behavioral or linguistic features. Fang et al. [5] have used a self multi-head attention-based convolutional neural network to detect fake news. They used CNN and self multi-head attention mechanism to extract the local n-gram semantic features to capture spatial relations between non-consecutive words and other methods [11, 14, 15, 20, 31].

Existing methods have been used in traditional discrete features, which can be sparse and fail to effectively encode the semantic information from the overall discourse. The above-specified methods failed to use global semantic contextual information to represent the document-level features for better spam review detection. Therefore, to mitigate the problems effectively, we propose the ACB model to capture complex representations and is expected to reveal more spamming activities with a higher accuracy rate and with less complexity.

3 Methodology

The Fig. 1 depict the proposed hierarchical neural network architecture Self attention-based CNN-BiLSTM (ACB) network which is designed in the Tensor flow framework with Keras APIs like word embedding layers, Convolutional layers, Max-pooling layer, Bidirectional Long Short Term Memory (Bi-LSTM) layers, self-attention layers, and dense layers. The input layer has a sentence length of 150 words where each word is passed to the word embedding layer and embedded into a 300-dimensional feature space. The output of this layer is the word vector matrix of size 150×300 which reflects the semantic distance and the relationship between words. The word embedding is a language modeling and feature learning technique in deep learning for NLP task, which maps the words in a vocabulary to vectors of numerical value [4, 30]. These vectors are low dimensional, continuous, and real-valued vectors. In this work, for performing the word embedding, the most influential Word2vec model is utilized. It preserves the syntactic and semantic relationship between words [17].

The self-attention mechanism is used to assign weightage to each word present in the sentence. A lower weight is assigned to lower impact features and a higher weight is assigned to higher impact features. The self-attention layer is added to identify the relative importance of all other words to a given word and identify the behavioral, relation-based, and linguistic features of the review. It looks for the internal connection within the words and extracts the relevant information in different presentation subspaces. This mechanism creates a context vector for each word which reflects the internal spatial relation (i.e., contextual relationship) between each word and the remaining other words. This results in a context vector that is associated with a given word. Thus, the output of the self-attention layer is a context vector matrix of size 150×300.

In the next layer, the context vector matrix is concatenated with the word vector matrix, results in the creation of an extended word vector matrix of size 150×600, and is fed to the CNN layer. CNN is used to learn continuous representations of a sentence

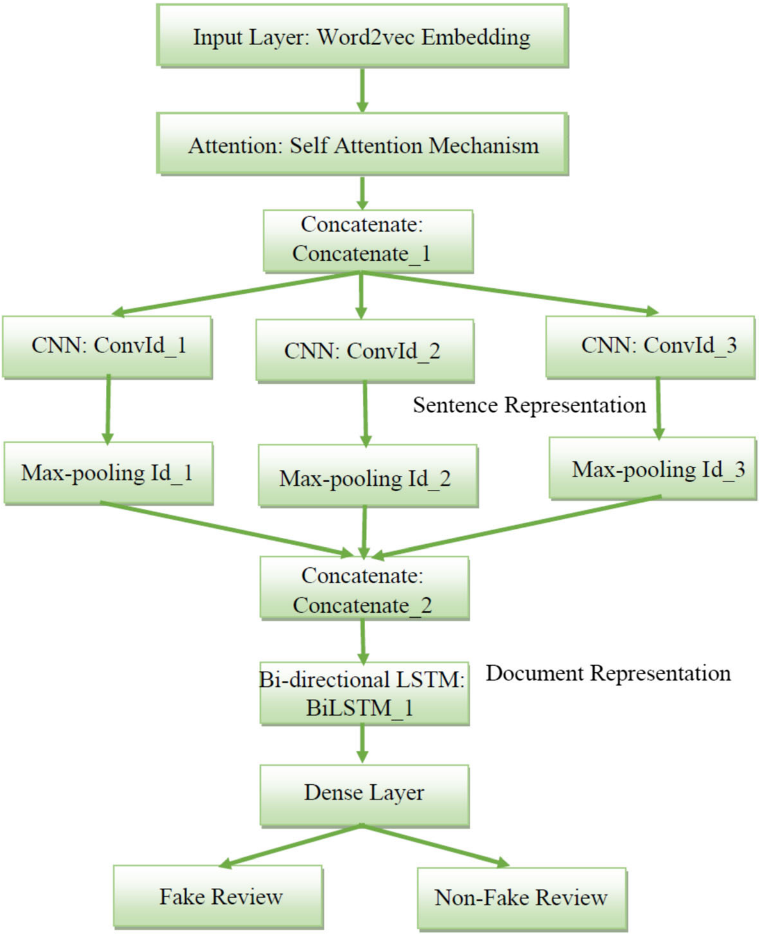


Fig. 1 The proposed ACB architecture model

and by applying three filters on the extended word vector matrix, the local n-gram semantics are extracted, and the sentence representation is generated. Following the convolution operation with a given filter Rectified Linear Unit (ReLU) non-linearity activation function is applied, and the feature element is obtained. By padding appropriately, the output feature vectors of all filters are obtained the same as of size 150. On each feature vector, a max-pooling operation is applied with a stride of 3 and the output is reduced to the feature vector of size 37. The output sentence vectors from the max-pooling layer are concatenated which results in a 37 × 6 matrix which becomes the input to the Bi-LSTM layer.

The regression technique has the dependent attributes with the produced logistic function which is utilized to discover the dependency within the features. This method is utilized for the dataset which consists of variables, spam or not. It generates the proficient feature vector that the prediction rate is computed using the procedure which lies within 0 to 1. If the value is around 1 means that the features have the highest amount of prediction else it has a minimized prediction rate, which is demonstrated in algorithm 1.

Algorithm 1 Generation of Prediction rate

```
Input: Feature subset, Dataset
Output: Prediction rate
Begin Procedure ( )
    Build the dataset $FS_{LR}$
    Every feature value has single point
    Split $FS_{LR}$ into $TrFS_{LR}$ and $TeFS_{LR}$
    Arrange the data illustration into the ascending order of $TrFS_{LR}$
    For every data illustration in $TrFS_{LR}$
        i= 1 to $\alpha_k$
            Compute the Logistics function as
                $LF = \alpha_0 + \alpha_1\delta_1 + \ldots + \alpha_k\delta_k$
            Compute the spam probability as
                $SP(a_i) = \frac{e^{LF}}{(1+e^{LF})}$
    End For
    For every data illustration in $TeFS_{LR}$
        Compute the generation function as
            $GF = \sum \beta_i \, SP(a_i) + \left((1-\beta_i)(1-SP(a_i))\right)$
    End For
        Compute the coefficients $\alpha_0,\ \alpha_1, \ldots + \ \alpha_k$ using the GF
        The coefficient values are updated as
            $SP(a_i) = \frac{e^{\alpha_0 + \alpha_1\delta_1 + \ldots + \alpha_k\delta_k}}{(1 + e^{\alpha_0 + \alpha_1\delta_1 + \ldots + \alpha_k\delta_k})}$
```

The Convolutional operation is computed with a mathematical formation to produce the 3rd function from the initial 2 functions. The Convolutional operation is represented in Eq. (1).

$$M = Z^{x\ x\ y} \tag{1}$$

Where M denotes the input matrix from the LSTM layer, Z demonstrates the real numbers, x denotes the length and y is the input matrix width. The Filter matrix (F) is computed in Eq. (2)

$$F = Z^{p\ x\ q} \tag{2}$$

Where p denotes the length and q demonstrates the width. The output matrix (O) is computed in Eq. (3)

$$O = Z^{g\ x\ r} \tag{3}$$

Where g denotes the length and r demonstrates the width. The Convolutional operation is constructed in Eq. (4)

$$c_{a,b} = \sum_{le=1}^{m}\sum_{wi=1}^{n} f_{\,le,wi} \bigotimes pi_{a+le-1,b+wi-1} \tag{4}$$

Where $c_{a,\,b}$ is the output matrix component, $f_{le,\,wi}$ is the weighted matrix and ⊗ demonstrates the cross multiplication within the elements. The source pixel is multiplied with the filter value to produce the destination pixel illustrated in Fig. 2. Equations (1) to (4) are used to produce the convolution operation through the LSTM layer and produce the output matrix.

The max pooling layer is utilized to reduce the feature map dimension using the aggregating data. The max pooling is pertaining to every element of the dataset, the operation is used to gather the needed feature by selecting the smallest value. The max pooling with the filters has been implemented to produce the spatial pooling which is illustrated in Fig. 3.

The dropout layer function is used to restrict the overfitting by using the parameter which falls within 0 to 1. It randomly eliminates the activation of the embedding layer that the dense illustration within a single neuron which is computed in Eq. (5).

$$fn(a,b) = \begin{cases} b & if\ a = 1 \\ 1-b & if\ a = 0 \end{cases} \tag{5}$$

Where a illustrates the expected results and b is the real value related probability elements for representation. Whenever the value of b is 1 the neuron having the real value will be eliminated and it is activated for other values, the entire representation of the dropout layer is illustrated in Fig. 4.

The entropy measurements denote the discrimination degree within ρ and τ where $\rho = \{\rho_1, \rho_2, \ldots, \rho_n\}$ and $\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$ are the two sets then the entropy $Ent(\rho, \tau)$ is computed in Eq. (6).

$$Ent(\rho,\tau) = \sum_{i=1}^{n} \rho_i \ln \frac{\rho_i}{\frac{1}{2}\ (\rho_i + \tau_i)} + (1-\rho_i) ln \frac{(1-\tau_i)}{\frac{1}{2}\ (\rho_i + \tau_i)} \tag{6}$$

The symmetric entropy is computed in Eq. (7).

$$Sym(\rho,\tau) = Ent(\rho,\tau) + Ent(\tau,\rho) \tag{7}$$

The output layer with the activation function like Softmax is used to calculate the probability of the input classes through the proposed technique. The input vector for providing the

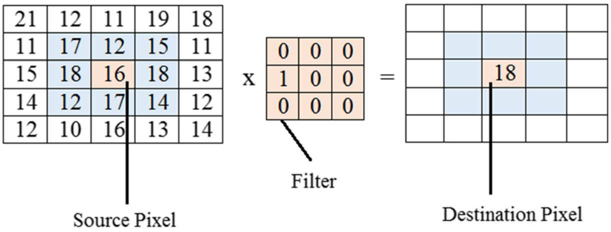


Fig. 2 Convolution operation

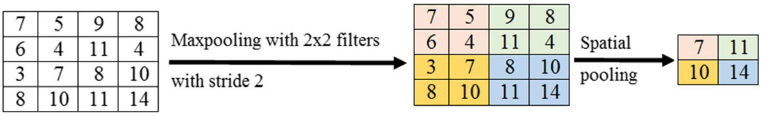


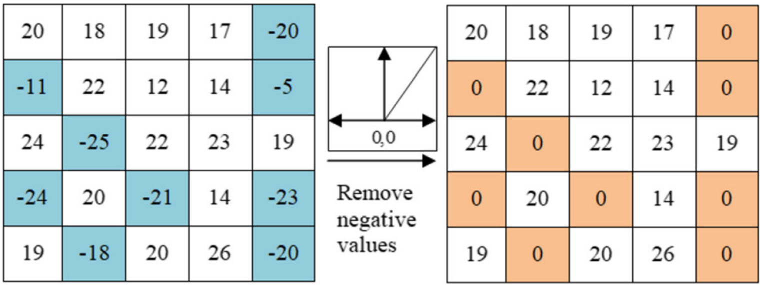
Fig. 3 Max pooling

classification using the Softmax function is computed using the weighted vector. The combined CNN with Bi-LSTM algorithm is used to identify the accuracy from the dataset using the dropout layer and embedding layer.

Algorithm 2 CNN and Bi-LSTM

```
Input: Dataset
Output: Accuracy
Begin Procedure ( )
    Initialize the batch size, filters, filters, input length, pool size;
    Include the maximum features, dimension into the embedding layer
    Append the input length into dropout layer
    Compute the parameters for LSTM layer
    Construct the Convolutional layer using filters, padding, ReLU activation function
    Produce the MaxPooling layer using pool size
    Compute the Flatten layer variables
    Activate the Softmax layer
    Execute compile function using the optimizer and entropy
    Accuracy is measured using the compile function
    For every epochs in the model
        Compute the fitness function using epoch, $X_{train}$ and $Y_{train}$ values
        Validating data using batch size
        Evaluate the model with $X_{train}$ and $Y_{train}$
        Compute the total accuracy
    End For
    Return accuracy
End Procedure
```

The Bi-LSTM layer learns the document composition and extracts contextual information with 37 units and outputs the forward and backward hidden state vectors at a given time step which are averaged instead of a concatenation so that the output of this layer is 37 × 6 matrixes. Similar to the work of Ren and Zhang [23], this matrix is flattened such that the output is a



20	18	19	17	-20
-11	22	12	14	-5
24	-25	22	23	19
-24	20	-21	14	-23
19	-18	20	26	-20

20	18	19	17	0
0	22	12	14	0
24	0	22	23	19
0	20	0	14	0
19	0	20	26	0

Fig. 4 Dropout layer operation

vector of size 222 which represents the features of the whole document. This vector is given as an input to a dense non-linear layer having 10 neurons with ReLU as an activation function, and a dropout of 0.25 is applied. The output of the first dense layer is a vector of size 10 which is connected to a second dense layer having a single node such that the output is either fake or non-fake review. Hence it is a binary classification problem, the objective of the training is to minimize the binary cross-entropy loss over the training dataset. *RMSprop* is used as an optimizer with a learning rate of 0.0001 and a decay of 10^{-6} in Fig. 5.

4 Experiments

In General, deceptive opinion spam detection is treated to be a classification problem. In this session, the experiments are conducted to empirically evaluate the efficiency of the proposed ACB model by applying it in spam review detection. The retrieved result is compared with other models to verify its performance level. The dataset used in the present analysis is obtained from YelpZip dataset (http://odds.cs.stonybrook.edu/yelpzip-dataset/) which consists of both truthful as well as deceptive reviews of 1,035,038 reviews and 458,325 reviewers. The dataset is divided into training, validation, and testing that Fig. 6 demonstrates the representation of the dataset split. The dataset consists of the subsets of the training set and the testing set, the testing set has the final estimation of performance. The training set has the annotated spam and non-spam reviews and the validation set to perform the tuning model. The testing set consists of unlabeled data that can be predicted as fake or truthful one.

It is a balanced dataset with 50% reviews obtained from MTurk, 25% from TripAdvisor, and the remaining from other sources. Out of the total reviews, 70% are used for training and validation and the remaining 30% are used for testing. Predicting an opinion review in the present dataset as a fake or non-fake is a binary classification problem. The confusion matrix for the binary classifier is defined in Table 1 where TP is True Positive, FP is False Positive, FN is False Negative, and TN is True Negative. Given a confusion matrix, it is possible to assess the performance of the model using metrics like Precision, Recall, F1-Score, Accuracy which are defined as follows:

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{8}$$

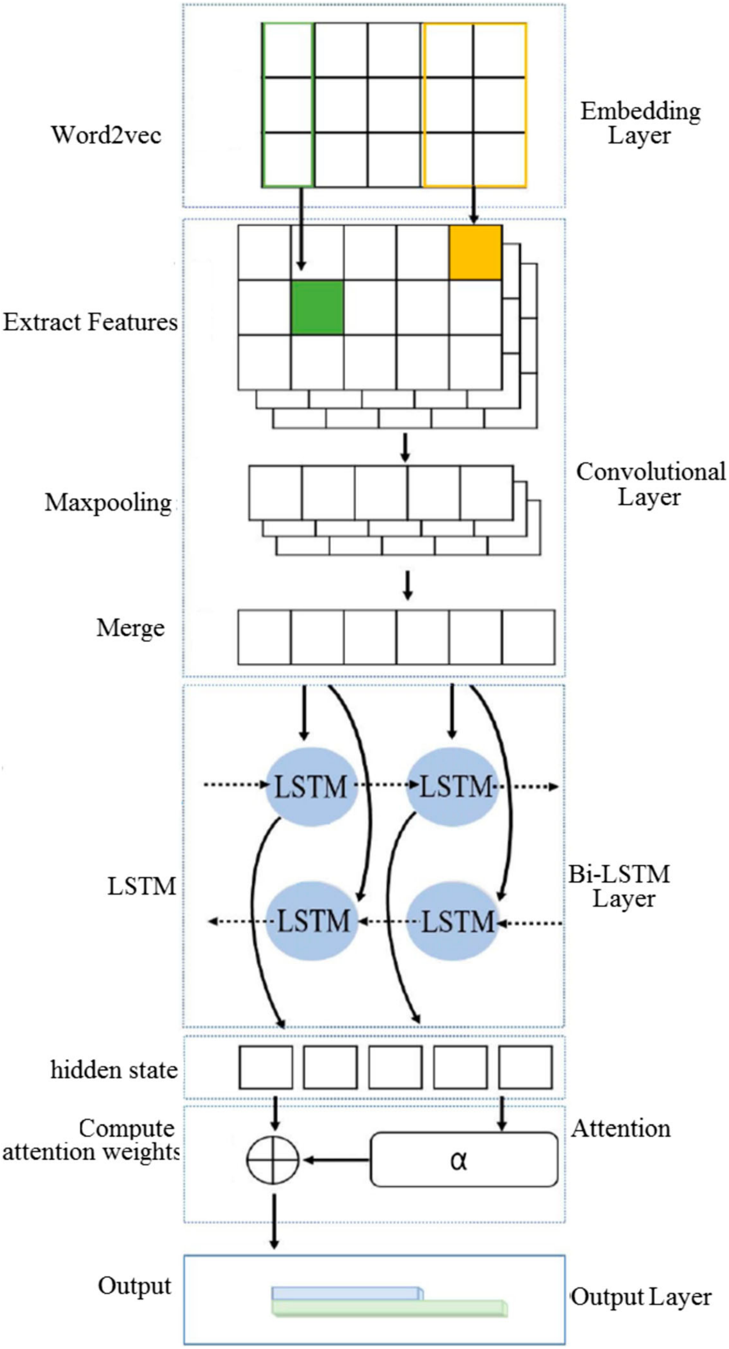


Fig. 5 Proposed layered architecture

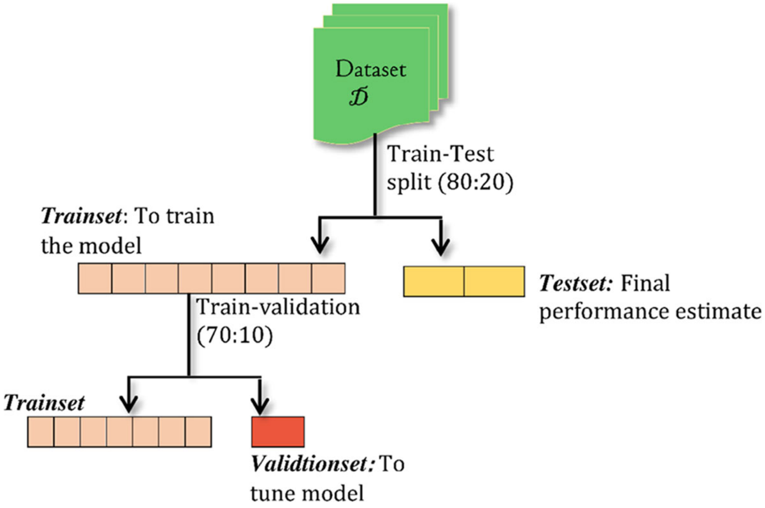

Dataset
$\mathcal{D}$
Train-Test split (80:20)
Trainset: To train the model
Testset: Final performance estimate
Train-validation (70:10)
Trainset
Validtionset: To tune model

Fig. 6 Dataset split

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F1\text{-}score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{11}$$

In Fig. 7, the loss functions which are evaluated over a set of samples for training and testing are plotted separately against the number of epochs. Weights of the network are randomly initialized, and both the training and the testing loss are achieved the same which is about 0.7 at the end of the first epoch. As a part of the minimization process, weights get updated after the entire batch passes through a forward pass. However, in the present case, a batch size of one is used. Both the training and testing losses are decreasing with the increasing number of epochs. After two epochs, the curve corresponding to a testing loss is steeper than the training loss. It indicates that the model is performing better on the testing dataset as compared to the training dataset up to six epochs. Thereafter, the curve corresponding to training loss is still showing a decreasing trend. On the other hand, the testing curve is

Table 1 Confusion matrix for opinion spam detection

	Ground truth fake	Ground truth non-fake
Predicted fake	TP	FP
Predicted non-fake	FN	TN

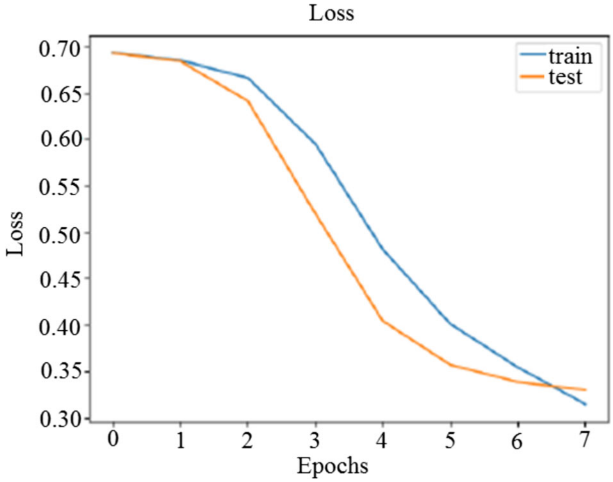


Fig. 7 The loss measure of the ACB model

flattening after six epochs. If the model is continued for training further after seven epochs, it will lead to overfitting. Thus, to avoid overfitting on testing data, early stop regularization criteria are followed, and the training process is stopped.

In Fig. 8, accuracy is plotted separately for training and testing samples against the number of epochs. The training accuracy is increasing steeply up to four epochs to a value of 0.77. Thereafter, the rate of increase in training accuracy is decreasing and at the end of the seventh epoch, accuracy reached a value of about 0.86. On the other hand, testing accuracy is increasing steeply up to the accuracy of 0.8 in the first three epochs, thereafter it is saturating to a value of 0.85 at six epochs. At the end of the seventh epoch, its value increased further to a value of 0.873.

The Receiver Operating Characteristic curve (ROC) graph gives a summary of the information available in the confusion matrices produced for each threshold without having to

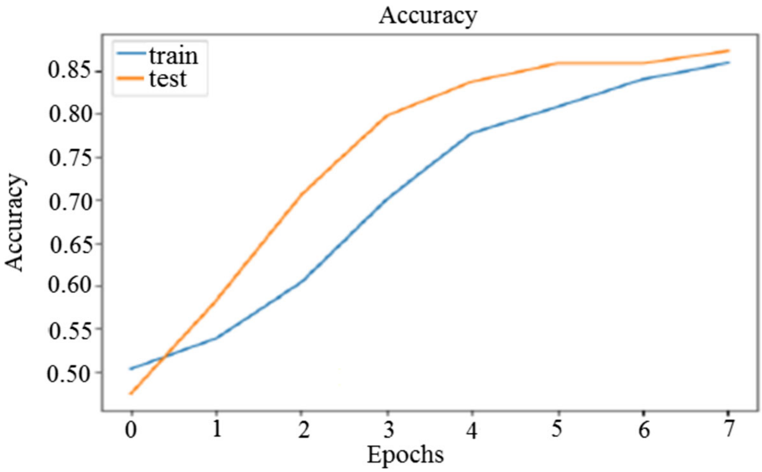


Fig. 8 The accuracy measure of the ACB model

calculate them. For plotting ROC, True Positive Rate (TPR) and False Positive Rate (FPR) need to be assessed at various thresholds as follows:

$$True\ Positive\ Rate\ or\ Recall = \frac{TP}{TP + FN} \tag{12}$$

$$False\ Positive\ Rate = \frac{FP}{FP + TN} \tag{13}$$

To classify it as fake or non-fake, we need to select some threshold value to say if the probability value is above the threshold, then it is considered as fake otherwise it is non-fake. Thus, by selecting a threshold value, TPR and FPR are evaluated, and it is represented as a point on the ROC curve. By selecting a certain number of threshold values, the ROC curve can be generated as shown in Fig. 9. It is to be noted that along the dotted line, TPR is equal to FPR which represents a random classification model. This dotted line is plotted just for clarity. The blue curve corresponding to the actual data of the present model. The curve is vertical for a certain level of threshold values, thereafter it is flattening. Based on the acceptable number of false positives, it is possible to select the right threshold value for the current model from this curve. Area Under Curve (AUC), is the area underneath the entire curve between points (0,0) to (1,0) which is about 0.936 for the present model. This indicates the higher potential of our proposed model in opinion spam detection.

Figure 10 shows the comparative performance of the models in terms of precision, recall, and F1-score. To evaluate the performance of our proposed model, we compare it to its variants such as Convolution Neural Network and Bi-LSTM (CNNB), Attention mechanism, and Convolution Neural Network (ACNN), standalone CNN, Hierarchical supervised Learning (HSL) and Spiral Cuckoo Search (Spiral CS). Since all the variants deal with a binary classification problem, the training objective is the same for all which is minimizing the binary cross entropy loss over the training dataset. RMSprop is used as an optimizer for all variants with a learning rate of 0.0001 and a decay of 10^{-6}. From the experimentation result, we notice that the proposed ACB model is doing well and it performs better with high accuracy in Fig. 11.

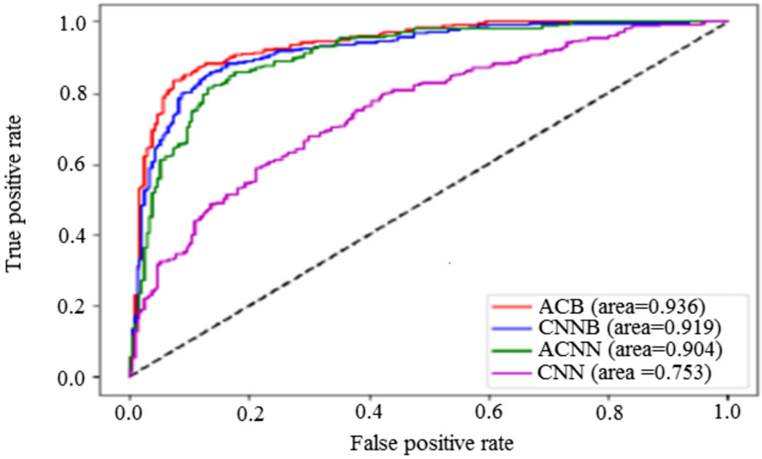


Fig. 9 Comparative ROC curves of the proposed model with its variants

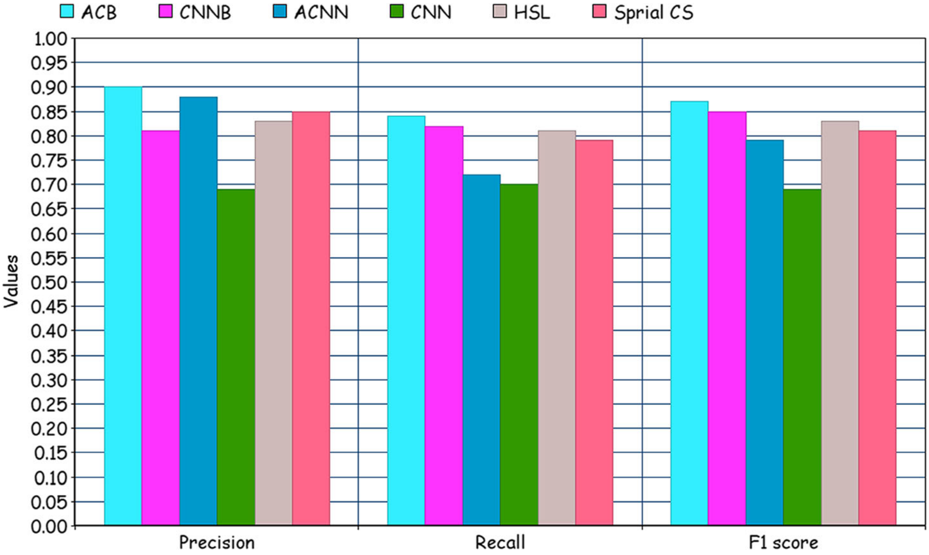


Fig. 10 Performance comparison

The computation time is computed as the time period required to finish the computational process for detecting the spam review from the dataset, the experimental results prove that the proposed ACB technique has the reduced amount of computation time compared with the related techniques of Spiral CS, HSL, CNN, ACNN and CNNB which is illustrated in Fig. 12.

The AUC score is used to identify the distributions from the gathered classifier that produces the significant result than the single distribution values. The mean map has the significant role for identifying the spam reviews that the combined distribution has been

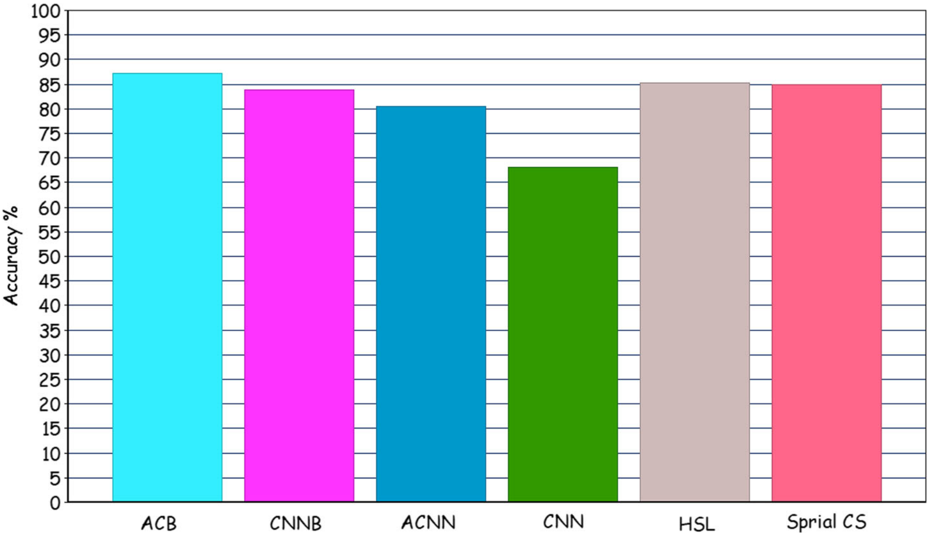


Fig. 11 Accuracy %

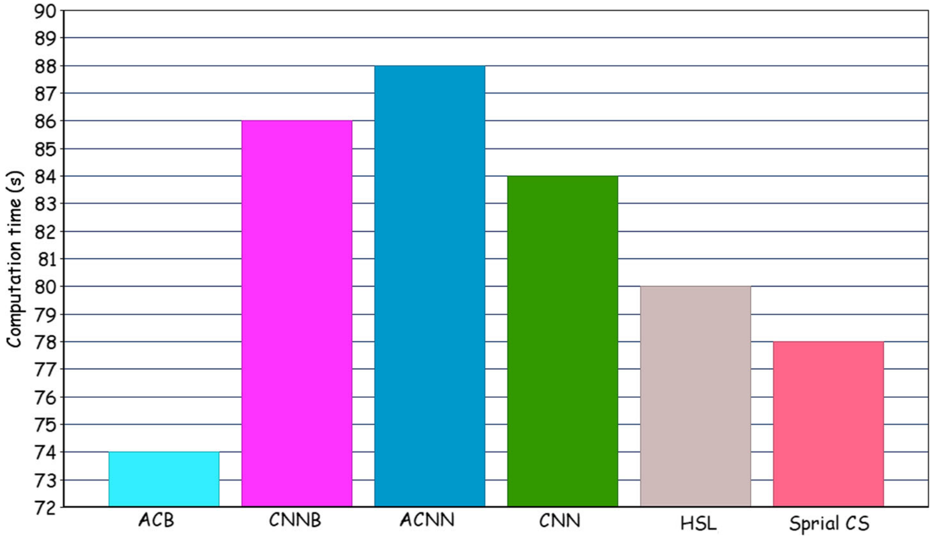


Fig. 12 Computation time

indicating the dependency within the user review and the overall rating into the specific time period. Figure 13 demonstrates that the proposed technique has the enhanced AUC score than other techniques.

5 Conclusions

In this work, we provide the novel architecture composed of self-attention mechanism, CNN, and Bi-LSTM for the identification of deceptive review in online portals. This model analyses

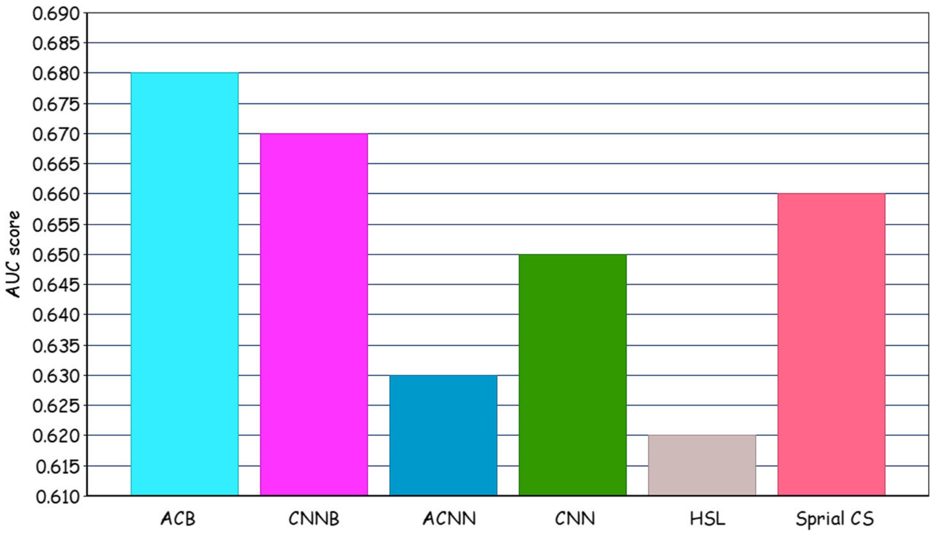


Fig. 13 AUC score

the words present in the entire document, preserves its global semantic contextual information and generates a document vectors as a representation of opinions. This representation is finally fed as an input to a dense layer to categorize as fake or non-fake reviews. Extensive experimental results demonstrate the effectiveness of our proposed approach. The comparative result show that our model outperforms better than the other variants. But till date, the problem of spamming review is open to researchers. Every proposed detection approach has certain drawbacks to identify all the harmful spam reviews. In future, we plan to perform spam detection based on aspect level with the rating deviation that the spam review detection will be applied in the fields of healthcare, marketing and law with efficiency technique to enhance into the analytical areas. The transfer learning related techniques have to be implemented for producing the multi-level prediction for providing the solution for the real-time issues.

References

1. Crawford, M., Khoshgoftaar, T. M., & Prusa, J. D. (2016, March). Reducing feature set explosion to facilitate real-world review spam detection. In The twenty-ninth international flairs conference.
2. Dong LY, Ji SJ, Zhang CJ, Zhang Q, Chiu DW, Qiu LQ, Li D (2018) An unsupervised topic-sentiment joint probabilistic model for detecting deceptive reviews. Expert Syst Appl 114:210–223
3. Drucker H, Wu D, Vapnik VN (1999) Support vector machines for spam categorization. IEEE Trans Neural Netw 10(5):1048–1054
4. Enríquez F, Troyano JA, López-Solaz T (2016) An approach to the use of word embeddings in an opinion classification task. Expert Syst Appl 66:1–6
5. Fang Y, Gao J, Huang C, Peng H, Wu R (2019) Self multi-head attention-based convolutional neural networks for fake news detection. PLoS One 14(9):e0222713
6. Feng S, Banerjee R, Choi Y (2012, July) Syntactic stylometry for deception detection. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers) (pp. 171-175).
7. Harris CG (2012, July) Detecting deceptive opinion spam using human computation. In Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence.
8. Heydari A, Ali Tavakoli M, Salim N, Heydari Z (2015) Detection of review spam: a survey. Expert Syst Appl 42(7):3634–3642
9. Horrigan J (2008) Online shopping. Pew Internet & American Life Project, Washington
10. Jindal N, Liu B (2008, February) Opinion spam and analysis. In Proceedings of the 2008 international conference on web search and data mining (pp. 219-230).
11. Kumar N, Venugopal D, Qiu L, Kumar S (Jan. 2018) Detecting review manipulation on online platforms with hierarchical supervised learning. J Manage Inf Syst 35(1):350380
12. Li L, Ren W, Qin B, Liu T (2015) Learning document representation for deceptive opinion spam detection, In Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data (pp. 393–404). Springer, Cham
13. Li L, Qin B, Ren W, Liu T (2017) Document representation and feature combination for deceptive spam review detection. Neurocomputing 254:33–41
14. Liu G, Fan D (2013) A model of visual attention for natural image retrieval. International Conference on Information Science and Cloud Computing Companion, Guangzhou, pp 728–733
15. Liu G-H, Yang J-Y, Li ZY (2015) Content-based image retrieval using computational visual attention model. Pattern Recogn 48(8):2554–2566
16. Luca M, Zervas G (2016) Fake it till you make it: reputation, competition, and yelp review fraud. Manag Sci 62(12):3412–3427
17. Mikolov T, Chen K, Corrado G, Dean J (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
18. Mukherjee A, Kumar A, Liu B, Wang J, Hsu M, Castellanos M, Ghosh R (2013, August). Spotting opinion spammers using behavioral footprints. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 632-640).
19. Ott M, Choi Y, Cardie C, Hancock JT (2011) Finding deceptive opinion spam by any stretch of the imagination. arXiv preprint arXiv:1107.4557.
20. Pandey AC, Rajpoot DS (Jun. 2019) Spam review detection using spiral cuckoo search clustering method. Evol Intell 12(2):147164

21. Rayana, S., & Akoglu, L. (2015, August). Collective opinion spam detection: bridging review networks and metadata. In Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining (pp. 985-994).
22. Ren Y, Ji D (2017) Neural networks for deceptive opinion spam detection: an empirical study. Inf Sci 385: 213–224
23. Ren Y, Zhang Y (2016, December). Deceptive opinion spam detection using neural network. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers (pp. 140-150).
24. Sedighi Z, Ebrahimpoor-Komleh H, Bagheri A, Kosseim L (2019, May). Opinion spam detection with attention-based neural networks. In the Thirty-Second International Flairs Conference.
25. Wang X, Liu K, Zhao J (2017, November). Detecting deceptive review spam via attention-based neural networks. In National CCF Conference on Natural Language Processing and Chinese Computing (pp. 866-876). Springer, Cham.
26. Wang CC, Day MY, Chen CC, & Liou, J. W. (2018, June). Detecting spamming reviews using long short-term memory recurrent neural network framework. In Proceedings of the 2nd International Conference on E-commerce, E-Business and E-Government (pp. 16-20).
27. Ye J, Akoglu L (2015, September) Discovering opinion spammer groups by network footprints. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 267-282). Springer, Cham.
28. Yoo KH, Gretzel . (2009, January). Comparison of deceptive and truthful travel reviews. In ENTER (pp. 37-47).
29. Zhang W, Du Y, Yoshida T, Wang Q (2018) DRI-RCNN: an approach to deceptive review identification using recurrent convolutional neural network. Inf Process Manag 54(4):576–592
30. Zhang L, Wang S, Liu B (2018) Deep learning for sentiment analysis: a survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8(4):e1253
31. Zhao Zhang, Zheng Lin, Jun Xu, Wenda Jin, Shao-Ping Lu, Deng-Ping Fan, Bilateral attention network for RGB-D salient object detection, arXiv 2020.
32. Zhao S, Xu Z, Liu L, Guo M, Yun J (2018) Towards accurate deceptive opinions detection based on word order-preserving CNN. Mathematical Problems in Engineering, 2018.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

P. Bhuvaneshwari[1] • A. Nagaraja Rao[1] • Y. Harold Robinson[2]

P. Bhuvaneshwari
thenameisbhuvanapatt@gmail.com

A. Nagaraja Rao
nagarajaraoa@vit.ac.in

[1] School of Computer science and Engineering, Vellore Institute of Technology, Vellore, India

[2] School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, India