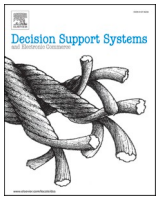
Decision Support Systems 166 (2023) 113911

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss



A deep learning approach for detecting fake reviewers: Exploiting reviewing behavior and textual information

Dong Zhang [a], Wenwen Li [b], Baozhuang Niu [a,*], Chong Wu [c]

[a] School of Business Administration, South China University of Technology, Guangzhou 510640, PR China
[b] School of Management, Fudan University, Shanghai 200433, PR China
[c] School of Economics and Management, Harbin Institute of Technology, Harbin 150001, China

ARTICLE INFO

Keywords:
Fake reviewer detection
Deep learning
Behavioral feature
Textual feature
Contextualized text representation

ABSTRACT

Ensuring the credibility of online consumer reviews (OCRs) is a growing societal concern. However, the problem of fake reviewers on online platforms significantly influences e-commerce authenticity and consumer trust. Existing studies for fake reviewer detection mainly focus on deriving novel behavioral and linguistic features. These features require extensive human labor and expertise, placing a heavy burden on platforms. Therefore, we propose a novel end-to-end framework to detect fake reviewers based on behavior and textual information. It has two key components: (1) a behavior-sensitive feature extractor that learns the underlying patterns of reviewing behavior; (2) a context-aware attention mechanism that extracts valuable features from online reviews. We rigorously evaluate each proposed module and the entire framework against state-of-the-art benchmarks on two real-world datasets from http://Yelp.com. Experimental results demonstrate that our method achieves state-of-the-art results on fake reviewer detection. Our method can be considered a tentative step toward lowering human labor costs in realizing automated fake reviewer detection on e-commerce platforms.

1. Introduction

Online consumer reviews (OCRs) play an essential role in assessing the quality of a product before consumers make informed decisions [1]. The past few years have witnessed increasing customer trust in OCRs [2]. According to a recent survey[1], nearly 80% of consumers trust OCRs as much as personal recommendations from friends or family, and more than 90% of consumers read OCRs before making a purchase decision.

However, as with many cases on the internet [3], fake online reviews are becoming increasingly prominent. An important reason is that the benefits of trading fake reviews are evident and proven. The Federal Trade Commission (FTC) points out that the outlay on fake reviews offers a 20 times payoff.[2] Therefore, firms or retailers have strong incentives to leverage fake online reviews to influence consumers, contributing to a booming market for fake online reviews. For example, in 2019, FTC found that Sunday Riley Skincare misled consumers by posting fake online reviews of its products for nearly two years.[3] Fake online reviews affect consumer trust and thus impact their purchase decision [4,5]. Besides, early fake online reviews negatively impact subsequent reviews [6]. In essence, fake online reviews are posted by fake reviewers (opinion spammers) who often exhibit anomalous behavior. Fake reviewer is the leading cause of misinformation on e-commerce platforms. Therefore, it becomes critical and urgent to develop effective methods to detect fake reviewers to maintain the authenticity of online reviews.

It is challenging to detect these reviewers due to the complexity of the reviewer's behavior and textual information. Prior studies have derived behavior-related and text-related features and fed them into machine learning approaches, including supervised classification [7,8] and unsupervised classification [9,10] to detect fake reviewers automatically.

Despite their important contributions to fake reviewer detection, there are still several limitations. First, although the importance of leveraging behavioral features in fake reviewer detection has been demonstrated [4,10], much of the research focuses on deriving novel behavioral features, which requires expensive human labor and

* Corresponding author.
E-mail addresses: zdscut@scut.edu.cn (D. Zhang), liwwen@fudan.edu.cn (W. Li), bmniubz@scut.edu.cn (B. Niu), wuchong@hit.edu.cn (C. Wu).

[1] https://www.brightlocal.com/research/local-consumer-review-survey/
[2] https://www.ftc.gov/news-events/press-releases/2011/03/firm-pay-ftc-250000-settle-charges-it-used-misleading-online
[3] https://www.ftc.gov/news-events/press-releases/2020/11/ftc-approves-final-consent-agreement-sunday-riley-modern-skincar

https://doi.org/10.1016/j.dss.2022.113911
Received 17 January 2022; Received in revised form 2 November 2022; Accepted 14 November 2022
Available online 21 November 2022

expertise. Second, in addition to behavioral features, text features, such as n-grams (bag of words) [11], part of speech n-grams [12], and word embedding [13], have been utilized to improve detection performance. However, these text features could negatively impact the detection performance of fake reviewers [8]. The bag of words (BoW) assumption considers a document as a bag of unordered words [14] and extracts features based on word frequency [15]. If an online review is full of informal words, abbreviations, and even obfuscated words, a feature vector for such a review is often very sparse and thus could negatively impact the detection performance. Linguistic features such as POS n-grams can be extracted from online reviews for fake reviewer detection. Such features may have difficulty detecting experienced fake reviewers. They attempt to sound convincing by using words or phrases that appear almost as frequently in genuine reviews as they do in fake reviews. They only overuse a small number of words in fake reviews, thus making them sound genuine. However, the small number of such words may not appear in every fake review, which explains why n-grams are less effective at classifying fake versus non-fake reviewers. Word embedding techniques such as Word2Vec capture limited semantic information because they leverage a static embedding vector for a word in different contexts. Such techniques may negatively impact detection performance when reviews contain words with different semantic meanings in different contexts.

To address the first challenge, the feature learning of behavioral features can be leveraged to improve detection performance. Feature learning is characterized by learning representations for specific tasks from raw data [16]. Compared with deriving novel behavioral features, feature learning requires less human labor, expertise, and can learn the underlying patterns of raw behavioral data. To address the second issue, we leverage the most advanced pre-trained language model, Longformer [17], to generate contextualized text representations from online reviews. Compared with traditional linguistic features, contextualized text representation can capture more semantic information from text inputs [18]. We then can utilize deep learning models to extract valuable features from the contextualized text representations and perform corresponding classification tasks. Therefore, we propose a novel deep learning-based framework for fake reviewer detection. The framework has two key novelties:

(1) We proposed a behavior-sensitive feature extractor that leverages the convolution filter to learn the underlying patterns of behavioral features.
(2) We design a novel context-aware attention mechanism, incorporating the most advanced pre-trained language model (Longformer) and other deep learning classifiers to extract valuable features from online reviews.

The remainder of this paper is organized as follows. We first review related work in section 2. Then in section 3, we detail the major components of our research design. Section 4 describes the experimental process, including the datasets, experiment design, and model evaluation. In section 5, we describe the practical and managerial implications of the proposed model. Section 6 discusses the main findings and future research directions.

2. Related work

2.1. Deception detection techniques

We classify the literature on deception detection techniques into two mainstreams: machine-learning and nonmachine-learning methods. The first stream of literature is built on information system design science, which leverages behavior and text features to identify suspicious reviewers. The second is based on methods outside the machine-learning context.

2.1.1. Machine learning methods

Our research aims to identify suspicious reviewers, of which the models are similar to fake review detection. Therefore, we review machine learning models in selected prior deception detection research, as shown in Table 1. In terms of fake reviewer detection, previous studies mainly focus on using methods with behavior features to identify suspicious reviewers. Existing studies on fake reviewer detection can be classified into three categories: (1) use raw behavior features only to detect suspicious reviewers, (2) leverage feature engineering for raw

Table 1
Summary of machine learning models in selected prior deception detection research.

Category	Year	Ref.	Behavior Feature	Text Feature	Models
Fake Reviewer Detection	2021	[13]	ANN for Raw features	Word embedding with word2vec	Graph model
	2019	[19]	Feature engineering on raw features	None	GMM, UM, FraudEagle, SpEagle, STK
	2018	[8]	Feature engineering on raw features	None	LR, k-NN, NB, AdaBoost, CART, RF
	2017	[20]	Raw	None	SVM, Coupled Hidden Markov
	2015	[9]	Raw	Linguistic Features	SpEagle
	2013	[10]	Raw	None	Author Spamicity Model
	2013	[21]	Raw	None	Markov Random Field model with Loopy Belief Propagation
	2012	[22]	Raw	None	SVM, LR, GSRank
	2021	[23]	None	Raw	BERT, DistilBERT, ALBERT, RoBERTa
	2021	[24]	None	Raw	BERT, DistilBERT
Fake Review Detection	2021	[25]	None	Word embedding	BiLSTM
	2020	[26]	Raw	Linguistic Features	SVM, LR, MLP, and NB
	2019	[27]	None	Word Embedding	A-BiLSTM
	2018	[28]	None	Word embedding	GANs
	2018	[29]	None	Word Embedding	LSTM
	2018	[30]	Raw	Word embedding	CNN, MLP
	2016	[31]	None	Word Embedding	ANN
	2011	[32]	Raw	Linguistic Features	LR, SVM, NB

Note: LR, Logistic Regression; SVM, Support Vector Machine; NB, Naive Bayes; k-NN, K Nearest Neighbor; GSRank, Ranking Group Spam; MLP, Multilayer Perceptron; SpEagle, SpaceEagle; CART, Classification and Regression Trees; RF, Random Forecast; ANN, Artificial Neural Network; CNN, Convolutional Neural Network; GRNN, Gated Recurrent Neural Networks; LSTM, Long Short-Term Memory; BiLSTM, Bidirectional Long Short-term Memory; A-BiLSTM, Attention-based BiLSTM; GANs, Generative Adversarial Networks; Adaboost, Adaptive Boosting; GMM, Gaussian Mixture Model; UM, Uniform Stacking; STK, Stacking; HAN, Hierarchical Attention Network; BERT, Bidirectional Encoder Representations from Transformers; DistilBERT, A Distilled Version of BERT; RoBERTa, Robustly Optimized BERT; ALBERT, A Light Version of BERT.

behavior features to obtain transformed values used for fake reviewer detection, and (3) combine behavior features and text features for fake reviewer detection. As shown in Table 1, existing studies mainly belong to the first two categories. Manaskasemsak et al. [13] and Rayana and Akoglu [9] are two typical exceptions that use both behavior and text features in fake reviewer detection (category 3). Manaskasemsak et al. [13] use ANN to deal with behavior and text features to create a reviewer node representation in the graph model and detect fake reviewers. Rayana and Akoglu [9] use raw behavior features and engineered text features in SpEagle (holistic approach) model to detect fake reviewers. However, Manaskasemsak et al. [13] and Rayana and Akoglu [9] use word embedding and engineered text features in their research. Such methods cannot capture dynamic semantic information.

Besides, both Kumar et al. [8] and Kumar et al. [19] have shown that feature engineering on raw behavior features can improve the detection performance of fake reviewers. However, feature engineering requires expensive human labor and expertise, which is tedious and time-consuming. This motivates us to leverage deep learning methods to extract high-level feature representations from raw behavior features. Since ANN cannot capture local dependencies among behavior features, we leverage CNN to extract high-level representations.

Existing studies on fake review detection can be classified into two categories: (1) leveraging text features only and (2) combing behavior and text features, as shown in Table 1. Most studies belong to the first category. For the first category, most studies use word embedding (e.g., word2vec) to deal with raw texts and feed them into machine learning classifiers, such as CNN and LSTM. Recently, several studies have leveraged transformers (e.g., BERT) to detect fake reviews [23,24,33,34]. For example, Mohawesh et al. [24] is an excellent work that applies BERT, DistilBERT, and RoBERTa for fake review detection. However, they do not combine behavior features. In addition, not all tasks can be easily represented by a transformer encoder architecture, requiring a task-specific model architecture to be added [35]. For the second category, prior studies mainly use raw behavior features with engineered text features and word embeddings. Existing research has rarely combined transformed behavior features and contextualized text representations from transformers to detect suspicious reviews.

2.1.2. Nonmachine-learning methods

Nonmachine learning methods are also used to distinguish fake reviewers from genuine users. Rayana and Akoglu [9] proposed a probabilistic graphic model to spot suspicious users based on metadata and relational data. Ye and Akoglu [36] introduced the Network Footprint Score (NFS) to measure the abnormality of spam campaign targets to identify spammer groups. Proudfoot et al. [37] demonstrated that coulometric behaviors (e.g., pupil dilation) are significant indicators in detecting deceivers.

2.2. Feature learning of behavioral features in deception detection

Prior studies mainly focus on leveraging feature engineering for behavioral features to improve classification performance. More specifically, extant research relies on creating new features or variable transformations [38]. For example, Mukherjee et al. [10] proposed several behavioral features (e.g., rating deviation, content similarity) to detect fake reviewers. However, the feature engineering approaches above require expensive human labor and expertise.

In contrast to feature engineering, feature learning (feature representation) refers to techniques that allow a system to automatically discover representations for classification from raw data [16]. As a typical feature learning technique, deep learning has received increasing attention and achieved remarkable success in many areas, such as computer vision [39] and natural language processing [40]. Some of the deep learning approaches have been used in deceptive detection. For example, Liu and Wu [41] combine hybrid deep learning approaches such as convolution neural network (CNN) and Text-CNN to detect fake news.

2.3. Text representation using deep architectures

As for text features, prior studies have mainly focused on simplistic linguistic features extracted from online reviews, such as n-grams [42] and the rate of misspellings [43]. Recently, a number of literature has appeared around the theme of pre-trained models (PTMs) that benefit from the Transformer architecture [44] for contextualized text representation, such as Bidirectional Encoder Representations from Transformers (BERT) [35], Roberta [45], and Longformer [17]. Before contextualized text representation, prior research mainly relied on static word embedding (e.g., word2vec) for text representation in deception detection. However, static word embedding techniques capture limited semantic information because the embedding vector of a word remains the same under different contexts.

Since deep contextualized representations contain more semantic information, researchers can leverage deep neural classifiers to perform specific tasks based on these representations. Such deep neural classifiers include CNN and Bidirectional Long Short-Term Memory (Bi-LSTM). Several existing studies have leveraged deep contextualized representations and deep neural classifiers to identify misinformation on social media [18]. For example, Kaliyar et al. [46] utilized BERT to create embedding vectors for input texts and then leveraged CNN to extract valuable information for detecting fake news.

2.4. Research gaps and questions

Our literature review revealed the following research gaps. First, in terms of dealing with behavior features, most studies use raw features directly or manual feature engineering on raw features, which could not capture local dependencies well among behavior features. Second, although several studies leverage transformers in deception detection, behavior features are rarely considered. Moreover, existing research leverages fine-tuning approach to adapt transformers to deception detection, which can be very computationally expensive. Compared with fine-tuning, the feature-based approach to transformer design provides great convenience and freedom for subsequent model design. Therefore, it is imperative to investigate whether combining behavior features with transformers using the feature-based approach could improve classification performance. Therefore, we propose the following research questions:

(1) Will high-level representations extracted from behavior features improve fake reviewer detection performance?

(2) Can text representations from the feature-based transformer adaptation method be utilized with transformed behavior features to improve detection performance?

3. Fake reviewer detection design

Our research design comprises three components: behavior-sensitive feature extractor, context-aware attention mechanism, and fake reviewer detection, as shown in Fig. 1. We detail the underlying process of each component in the following subsections.

3.1. Behavior-sensitive feature extractor

In this section, we assume that behavior features are locally dependent. We have the following considerations. First, fake reviews are written not solely to impact the product's star rating, but also to influence the desire of consumers to buy the product. Fake reviewers attempt to manipulate their behaviors to act just like genuine reviewers, and thus can make people trust their reviews. For example, fake reviewers behave as genuine reviewers in behaviors such as review count [20]. Accordingly, fake and genuine reviewers have similar behaviors. If these features are assumed to be independent, this may affect the performance of

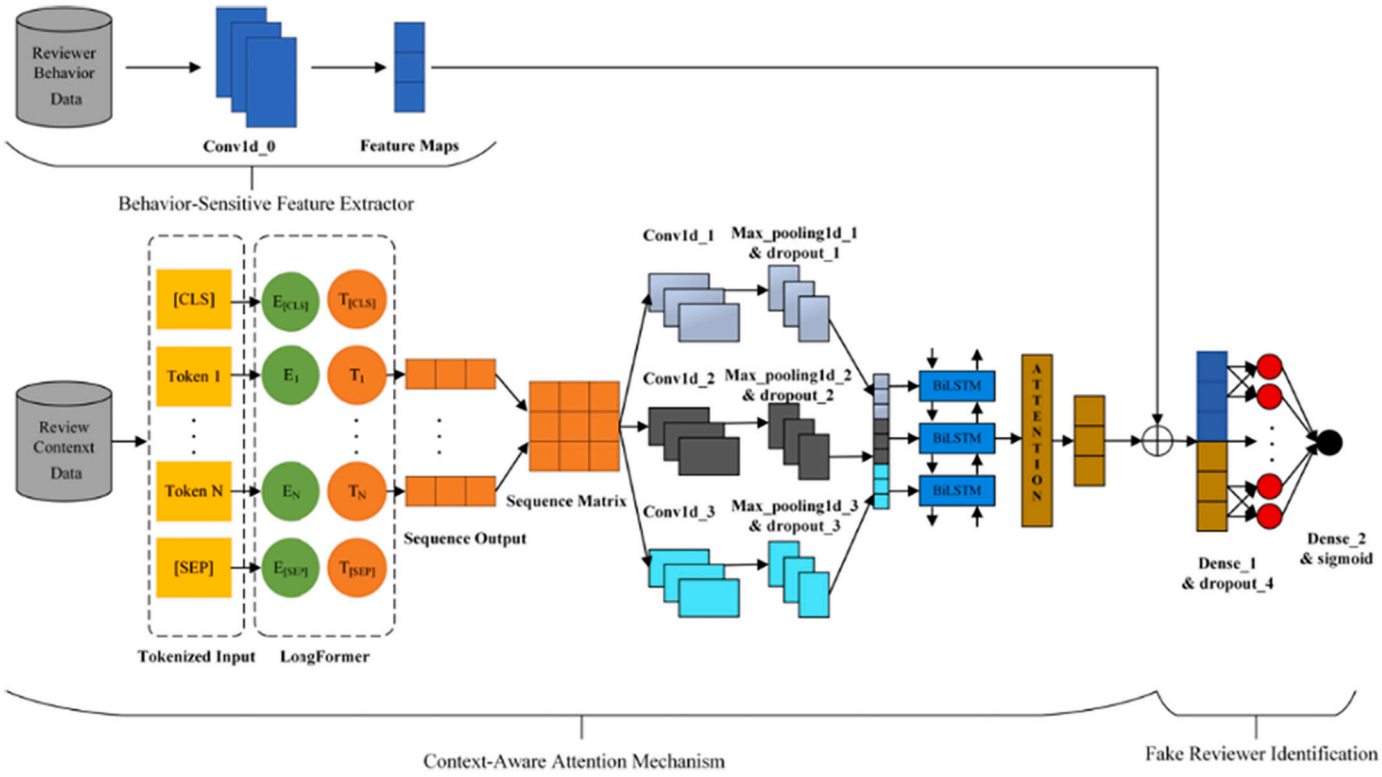


Fig. 1. Proposed fake reviewer detection design.

the fake reviewer detection. In view of this, previous studies [8,38] have claimed that incorporating interactions among features could improve fake review/reviewer detection performance. Based on the above considerations, we argue that it could improve the classification performance if the behavior features are assumed to be locally dependent in determining whether a review is fake or not. For the sequence of the behavior features, we argue that highly correlated features are likely to be more locally dependent than those that are less correlated. We use Spearman's rank correlation to evaluate the relationship between behavior features, and these coefficients are leveraged to rank those

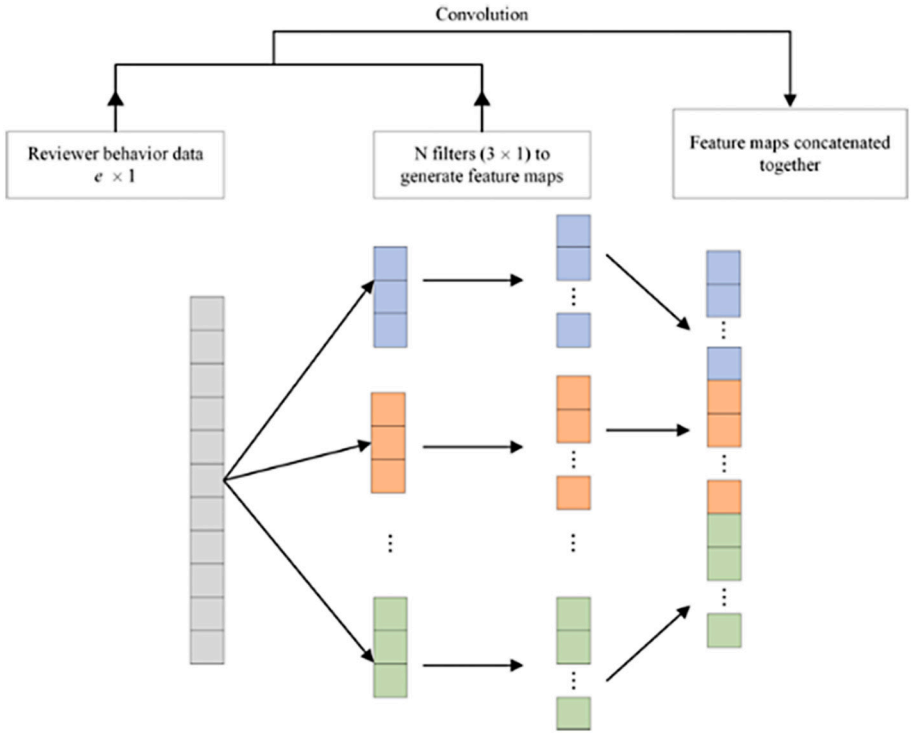


Fig. 2. The architecture of behavior-sensitive feature extractor. The input is a sequence of e-dimension vector. We apply n filters (3 × 1) to generate the corresponding feature maps. Finally, the n maps are concatenated together to generate a single feature vector.

behavior features used in the CNN. We have conducted additional experiments to demonstrate the efficacy of the selected behavior feature list. Please see appendix C for more details.

We leverage the one-dimensional convolution filter to extract features from reviewer behavior data. We use the convolution filter because of its strength in extracting local dependency from behavioral features. Fig. 2 provides a schematic demonstrating the proposed architecture for behavioral feature learning.

The behavioral data of each reviewer is expressed as a vector and we demote the vector's dimensionality by 1. Hence, the dimensionality of behavior data of each reviewer is $e \times 1$. We then adopt the 1-d convolution layer with filter size 3×1 to generate feature maps. we denote the weight matrix of a filter by $W_{rb} \in \mathbb{R}^{3\times 1}$ and the vector of reviewer behavior by $X_{rb} \in \mathbb{R}^{e\times 1}$, where the subscript *rb* denotes reviewer behavioral data. Then, we can generate the feature map m_i^{rb} with the following convolution operator:

$$m_i^{rb} = f\left(W_{rb}^i \cdot X_{rb} + b_{rb}\right), \quad i \in 1, 2, \ldots, n \tag{1}$$

where f is an activation function, b_{rb} is a bias term, and n is the number of filters. We use rectified linear unit (RELU) function as the activation function due to its better performance than other functions like sigmoid. We concatenate the outputs of convolution layer and generate the feature maps $M^{rb} = [m_1^{rb}, m_2^{rb}, \ldots, m_n^{rb}]$ that are used as the input of fake reviewer identification layer.

3.2. Context-aware attention mechanism

Transformers are typically used for downstream tasks in two main ways: feature-based and fine-tuning. In this research, we leverage the feature-based approach instead of fine-tuning approach with the following considerations:

(1) there are significant computational benefits for the feature-based approach because it computes an expensive representation of the training data once and then runs many experiments with cheaper models on top of this representation.

(2) not all tasks can be easily represented by a transformer encoder architecture and therefore require a task-specific model architecture to be added [35]. As the source of pre-training and target data differ, a potential domain gap might exist between pre-training and target tasks. Therefore, adding a task-specific model architecture could increase the detection performance of suspicious reviewers. Since the average review length per reviewer is higher than 512, we take Longformer for review text embedding. For the task-specific architecture, we leverage CNN, BiLSTM, and attention mechanism to extract high-level representations as they are widely used in deception detection research [24]. The Longformer is used to obtain contextualized text representations for capturing valuable information from input texts. Although BiLSTM has been extensively used in dealing with sequence modeling problems, it is not capable of extracting local context information and cannot emphasize the critical parts of the contextual information [47]. Therefore, we apply CNN before BiLSTM to extract local features and reduce the dimensionality of the embedding vectors. We then adopt the attention mechanism to assign different weights to the outputs of the BiLSTM and focus on the important parts of the context.

3.2.1. Pre-trained models in NLP

In this paper, we leverage Longformer proposed by Beltagy et al. [17] with the following considerations: (1) Longformer improves BERT's architecture by enlarging the training volume and adopting an efficient attention mechanism, (2) we combine the large amounts of review content and behavior data associated with each user, as suggested in [8,48]. Therefore, the aggerated review content has more words. The average number of words per reviewer is 682, more than 512. So, we use Longformer instead of other transformers in this study.

3.2.2. CNN layer

We utilize CNN's strength in local dependency extraction to extract n-gram features from contextualized text representations. In terms of text data, we utilize Longformer to convert one sentence to a sentence matrix that is represented by $X_{rc} \in \mathbb{R}^{s\times d}$, where s denotes the number of tokens and d is the dimension of the embedding vector for each token. We denote the weight matrix of j-th filter by $W_j^{rc} \in \mathbb{R}^{h\times d}$, where h denotes the kernel size, $j \in [1, n]$, and $h \in [1, m]$. Then, we apply the Eq. (1) on $X^{s\times d}$ and generate $m_i^{rc,\,j}$ as follows:

$$m_i^{rc,j} = f\left(W_j^{rc} \cdot X_{[i,i+h-1]} + b_{rc}^j\right), \quad 1 \le i \le s - h + 1 \tag{2}$$

Where $X_{[i,i+h-1]}$ denotes the sub-matrix of X from row i to row j, b_{rc}^j is a bias vector. W_j^{rc} is the weight matrix, and f is the RELU function. All feature maps extracted by the j-th filter W_j^{rc} can be represented as $M^{rc,\,j} = [m_1^{rc,\,j}, m_2^{rc,\,j}, \ldots, m_{s-h+1}^{rc,\,j}]$.

Following convolution layer, the pooling layer is to reduce the size of feature maps and prevent overfitting. Max-pooling is used to select important features by max values, whose max-pooling operation is presented as below:

$$M_h^{rc,j} = max\left\{m_1^{rc,j}, m_2^{rc,j}, \ldots, m_{s-h+1}^{rc,j}\right\} \tag{3}$$

We concatenate the outputs of max-pooling layer and generate the pooled feature maps $d_h = (M_h^{rc,\,1}, M_h^{rc,\,2}, \ldots, M_h^{rc,\,n})$ for each region size h. Therefore, after the CNN layer, the sentence representation becomes $D = (d_1, d_2, \ldots, d_l)$, where l denotes the number of region size. And, we fed D to BiLSTM-Attention Layer.

3.2.3. BiLSTM -attention layer

Since traditional (unidirectional) LSTM can only capture information from previous context, we leverage the bidirectional LSTM (BiLSTM) to capture two direction dependencies, forward LSTM and backward LSTM. Fig. 3 presents the overall BiLSTM with attention mechanism.

As shown in Fig. 3, BiLSTM is comprised of two subnetworks, $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$. The output of the input is computed as follows:

$$\overrightarrow{h}_t = \overrightarrow{LSTM}(d_t), \quad t \in [1, l] \tag{4}$$

$$\overleftarrow{h}_t = \overleftarrow{L}\ STM(d_t), \quad t \in [l, 1] \tag{5}$$

$$h_t = \sigma\left(\overrightarrow{h}_t, \overleftarrow{h}_t\right) \tag{6}$$

where σ, a concatenating function, is used to deal with $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$.

The attention mechanism is used to assign different weights to tokens contributing differently to the text classification, which has achieved great success in a variety of NLP tasks. By adopting attention mechanism on the top of a BiLSTM, the attention model can highlight the related information and suppress the irrelated information. The output S of the attention layer is computed as follows:

$$u_t = tanh(W_s h_t + b_s) \tag{7}$$

$$\alpha_t = \frac{exp(\nu^T \cdot u_t)}{\sum_t exp(\nu^T \cdot u_t)} \tag{8}$$

$$S = \sum_t \alpha_t h_t \tag{9}$$

where W_s and b_s are the weight matrix and bias matrix. ν is the coefficient matrix of the attention layer.

3.3. Fake reviewer detection

We concatenate the outputs of the behavior-sensitive feature

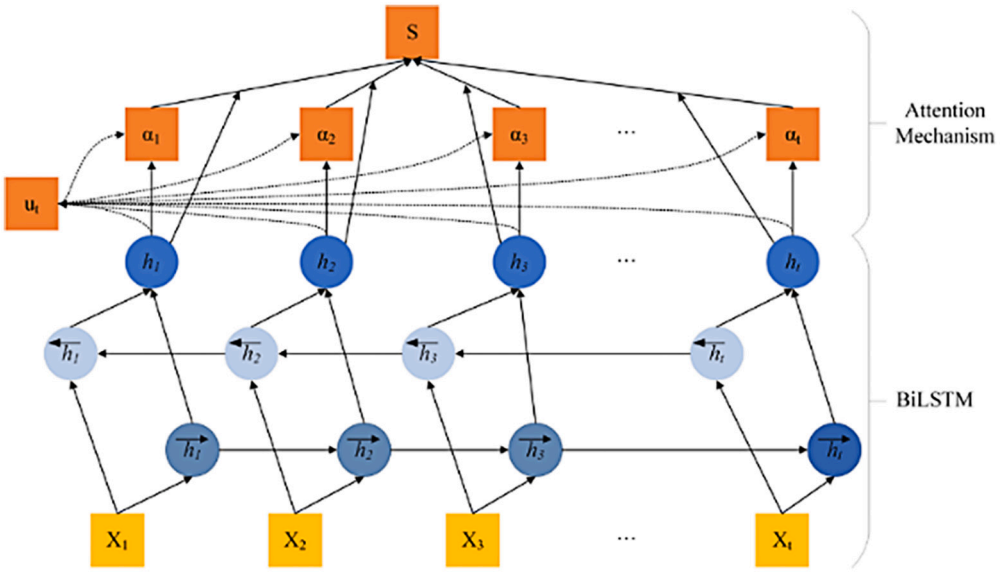


Fig. 3. BiLSTM layer and attention layer.

Note: X_t is the t-th input, $t \in [1,n]$, $\overrightarrow{h_t}$ is the forward hidden state, $\overleftarrow{h_t}$ is the backward hidden state, h_t is the output of BiLSTM layer, u_t is the hidden representation, α_t is the attention weights to h_t, and S is the output of attention layer.

extractor and context-aware attention mechanism. Then, a fully connected layer receives the concatenated information and classifies the reviewer as fake or genuine. We use a sigmoid function to select the most probable predefined label for a reviewer.

4. Method evaluation

4.1. Data testbed

We use data from http://Yelp.com that has been used by previous research [4,19,49]. More specifically, we use the YelpZIP dataset shared by Rayana and Akoglu [9], which has also been adopted by many studies [19,50].

The YelpZip dataset has 260,227 users who wrote 608,598 reviews between July 2010 and November 2014, with 80,466 fake reviews and 528,132 genuine reviews. Since one reviewer can post multiple reviews, we use Fig. 4 to show the empirical distribution corresponding to the number of reviews written by each user. It is apparent from Fig. 4 that the distribution is highly skewed in nature, and most users write fewer than 20 reviews.

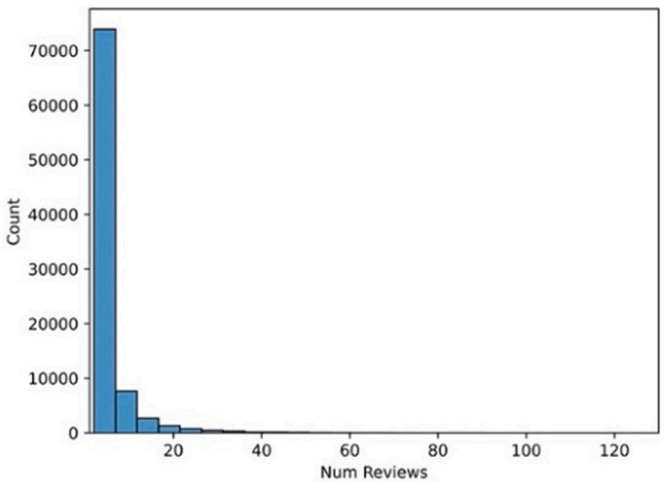


Fig. 4. Empirical Distribution of Number of Reviews Written by Each User.

4.1.1. Data pre-processing

First, we remove inactive reviewers and restaurants from our data set. Specifically, we remove reviewers who have written fewer than three reviews. We then remove reviewers whose labels are inconsistent because Yelp labels some reviewers as both fake and genuine. In addition, we filter restaurants with less than three ratings in our data set.

Several behavioral features for detecting opinion spammers have been proposed in prior studies [9,10,21]. Please see appendix D for more details on these behavior features. Table 2 shows the summary statistics for the behavioral features in YelpZIP. In addition, this study aims to investigate the detection performance improvement with simplistic features versus contextualized text representations. Hence, we leverage textual features extracted from review content by previous studies [4,9], as shown in Table 3.

4.2. Experiment design and performance metrics

To evaluate our proposed method, we conduct two main categories of experiments. Experiment 1 evaluates the detection performance of our proposed behavior-sensitive feature extractor versus state-of-the-art benchmarks using behavioral data. These benchmarks include logistic regression (LR), random forest (RF), classification and regression trees (CART), support vector machine (SVM), and naive Bayes (NB), which have been wildly used in prior research [4,8]. Experiment 2 compares the detection performance improvement using simplistic text features and behavioral features versus features extracted from the context-

Table 2
Feature statistics of the filtered dataset.

Feature	Mean	Std	Min	Max
Review Count	3.99	5.61	2	159
User Tenure	274.98	448.03	0	3177
Review Gap (Avg)	124.32	237.65	0	2979
Review Gap (Std)	46.56	108.82	0	1215
Rating Entropy	0.86	0.57	0	2.32
Rating Deviation (Avg)	0.86	0.48	0	3.26
Rating Deviation (Std)	0.41	0.29	0	1.76
Time of Review (Avg)	1492.71	698.45	0	3602.50
Time of Review (Std)	549.70	366.88	0	1714.50
Rating Scores (Avg)	3.94	0.82	1	5
Rating Scores (Std)	0.65	0.54	0	2

Table 3
Text-based features adopted by prior studies.

Feature	Description
Review length	The number of words in reviewer's comments
Average sentence length	Average number of words per sentence.
Subjective	The ratio of subjective words to objective words
Average similarity	Average pairwise similarity among users' reviews
Max similarity	The maximum similarity among all review pairs
Noun ratio	Percentage of nouns
Pos counts	Number of nouns, verbs, adjectives, personal pronouns, and pronouns
Lexical validity	The number of misspellings to the total number of words
Sentiment orientation	The ratio of positive and negative words to total number of words
Lexical diversity	The ratio of unique words to total number of words
Content diversity	The ratio of nouns and verbs to the total number of nouns and verbs
Bigram-diversity	The ratio of unique bi-grams to the total number of pos bigrams
Redundancy	The ratio of repeated words to the total number of words
Self-reference diversity	The ratio of first-person pronouns to the total number of pronouns
Capitalized diversity	Percentage of all-capitals words

aware attention mechanism. In experiment 2, we include more stronger baselines, including CNN, BiLSTM, C-LSTM, BERT, DistilBERT, RoBERTa, ALBERT, and Longformer, used by prior studies [23,24,34]. Note that for neural network models (CNN, BiLSTM, C-LSTM), we leverage pre-trained GloVe embedding methods with 100-dimensions.

4.2.1. Evaluation metrics and experiment environment

We perform k-fold cross-validation (k = 5) and use four well-known performance metrics, including precision (P), recall (R), F_1-score (F_1), and area under curve (AUC) scores to measure classification performance, which have been widely used in previous research [2,4,8]. The metrics of *P*, *R*, and *F* are defined in Eqs. (10)–(12).

$$P = \frac{TP}{TP + FP} \tag{10}$$

$$R = \frac{TP}{TP + FN} \tag{11}$$

$$F_1 = \frac{2PR}{P + R} \tag{12}$$

where TP denotes the number of faker reviewers that the classifier correctly predicts, FP is the number of genuine users that are incorrectly classified as fake reviewers, TN is the number of correctly detected genuine users, and FN is the number of fake reviewers that are incorrectly classified as genuine users.

4.3. Results and discussion

4.3.1. Model performance

We present the evaluation results of experiments 1 and 2 in Table 4. As Table 4 shows, when textual features are not used (i.e., the benchmarks), the proposed behavior-sensitive feature extractor outperforms other baselines in accuracy (85.05%), F_1-score (0.7198), and the AUC score (0.7924). To gain a better insight into the performance improvement of our proposed model, we conducted the pair-wise *t*-test, based on five performance samples of a model produced by 5-fold cross-validation, to test the performance obtained from the behavior-sensitive feature extractor and other benchmarks, as shown in Table 5.

Table 5
Top 15 words contributing to $\triangle KL$ with their fake/non-fake class probabilities.

Word	ΔKL_{word}	$P(w\|F)$	$P(w\|N)$
food	3.3910E-03	1.3992E-02	1.2299E-02
great	2.7517E-03	9.5786E-03	8.2055E-03
service	2.3735E-03	6.4492E-03	5.2665E-03
best	1.6327E-03	4.6363E-03	3.8225E-03
pizza	1.5818E-03	3.9915E-03	3.2037E-03
restaurant	1.3873E-03	5.5319E-03	4.8393E-03
place	1.2255E-03	1.2577E-02	1.1965E-02
always	1.1264E-03	2.9615E-03	2.4004E-03
amazing	1.0071E-03	3.0548E-03	2.5526E-03
staff	8.3979E-04	2.4495E-03	2.0308E-03
love	8.2081E-04	3.6392E-03	3.2293E-03
time	8.0374E-04	6.8515E-03	6.4497E-03
favorite	7.1514E-04	2.1495E-03	1.7929E-03
excellent	6.9266E-04	1.8114E-03	1.4663E-03
delicious	6.6110E-04	4.6012E-03	4.2708E-03

Note: E-02, 10^{-2}; E-03, 10^{-3}; E-04, 10^{-4}.

Table 4
Performance comparison of models with all features vs. behavioral features using YelpZIP.

Experiment	Algorithm for BF	Algorithm for Text	Acc	Precision	Recall	F_1-score	AUC
1 (Behavioral features only)	LR	None	76.35%*	0.6890***	0.6822*	0.6855**	0.7030***
	RF		77.34%*	0.7008*	0.6781*	0.6892*	0.7033***
	SVM		77.39%	0.7149*	0.6850*	0.6996*	0.7119***
	CART		73.25%**	0.6902**	0.6766**	0.6833**	0.6995***
	NB		76.38%*	**0.8043**	0.6409***	0.7134	0.7093***
	Conv1d		**80.51%**	0.7356	**0.7047**	**0.7198**	**0.7924**
2 (All features)	LR	LR	76.85%***	0.7354	0.6998**	0.7171**	0.7106***
	RF	RF	77.37%***	0.7500	0.6994**	0.7238**	0.7149***
	SVM	SVM	76.80%***	0.7563	0.7000***	0.7270**	0.7174***
	CART	CART	72.08%***	0.7433*	0.6988***	0.7203**	0.7123***
	NB	NB	75.32%***	**0.9016**	0.5931***	0.7155**	0.6941***
	Conv1d	CNN	77.45%***	0.6475	0.6745*	0.6581*	0.7929*
		BiLSTM	78.01%***	0.7249	0.7161*	0.7205*	0.7954*
		C-LSTM	78.10%***	0.7225	0.7224*	0.7224*	0.7991*
		BERT	78.67%***	0.7225*	0.7132*	0.7178*	0.8052*
		ALBERT	79.90%***	0.7259	0.7186**	0.7221*	0.8034**
		DistilBERT	79.55%***	0.7231*	0.7276	0.7252*	0.8048*
		RoBERTa	79.82%***	0.7296	0.7226	0.7260	0.8075*
		Longformer	80.66%***	0.7256	0.7340	0.7297*	0.8014*
	Our model		**85.05%**	0.7689	**0.7643**	**0.7664**	**0.8358**

Note: *, **, and *** denote significance level at the 0.05, 0.01, 0.001 levels, respectively. Numbers in Bold are the best performance in each category. BF, Behavior Features; LR, Logistic Regression; RF, Random Forest; SVM, Support Vector Machine; CART, Classification and Regression Tree; NB, Naïve Bayes; Conv1d, one-dimensional convolution; BERT, Bidirectional Encoder Representations from Transformers; DistilBERT, A Distilled Version of BERT; RoBERTa, Robustly Optimized BERT; ALBERT, A Light Version of BERT. Longformer, The Long-Document Transformer.

The statistical tests show that the proposed architecture significantly outperforms other benchmarks in accuracy, recall, F1-score, and AUC.

The behavior-sensitive feature extractor extracts local dependencies from behavior features and generates new feature maps for classification. Instead of using raw behavioral features, the behavior-sensitive feature extractor considers the possible interactions among behavioral features, which leads to higher classification performance in fake reviewer detection.

Besides, when leveraging features extracted from the context-aware attention mechanism, the proposed model improves the classification performance by 6.4% (from 0.7198 to 0.7664) of F_1-score and 5.5% (from 0.7924 to 0.8358) of AUC. The result suggests that the proposed context-aware attention mechanism improves classification performance. Also, we conducted the pair-wise *t*-test to test the performance of the proposed model and other benchmarks using all features, as shown in Table 4. The statistical tests (Table 4) show that the proposed model significantly outperforms other benchmarks in accuracy, F1-score, and AUC score.

4.3.2. Feature importance analysis

To compare the importance of behavior and text data, we evaluate our model on the test set to obtain the respective feature importance. Particularly, we use permutation importance to measure performance, which evaluates the performance by monitoring the change in the relevant metric when each feature is randomly permuted [51]. Permutation importance benefits from being compatible with black-box models and is less sensitive to noise. We present the importance results of our model in Fig. 5. Interestingly, the text is the fourth important feature for identifying suspicious reviewers. This suggests that text contains valuable information for detecting fake reviewers, even with other variables. Besides, from Fig. 5, behavior features are more effective in detecting suspicious reviewers. One possible explanation is that fake reviewers are more experienced in manipulating review content, which makes the text feature less effective. However, the behavior features are challenging to fabricate and more effective for detecting fake reviewers.

4.3.3. Analysis of text content

We leverage SHAP [52] to show which words contribute to authentic and fake reviews. Fig. 6(a) and 6(b) show the top 15 important words for cases where text contributes to authentic and fake reviews, respectively. The bar length represents a word's impact in the test bed.

Fig. 6 can provide insights for managers of digital platforms. First, fake reviews have more verbs than nouns. For example, Fig. 6 (b) shows that words such as "go", "eat", and "try" have a significant impact on detecting fake reviewers. A possible explanation for this might be that fake reviewers replace actual insights with pleasant or sounding stories that are meant to give the readers a positive impression. Second, fake reviews include more positive words. For example, our model focuses on words such as "better", "great", and "perfect" in detecting fake reviewers. This result may be explained by the fact that fake positive reviews are more common than fake negative reviews. For suspicious reviewers, it is easier and less risky to affect overall ratings by posting positive reviews than by posting a series of negative reviews against many competitors. We next compare the word distribution between fake and authentic reviews to reveal some interesting insights.

As for the difference of word distribution between fake and authentic reviews, we leverage Kullback-Leibler (KL) divergence: $KL(F \parallel N) = \sum_i F(i) log\left(\frac{F(i)}{N(i)}\right)$, where $F(i)$ and $N(i)$ are the respective probabilities of word in fake and non-fake reviews [53]. KL provides a quantitative estimate of how much do fake reviews linguistically differ from non-fake reviews. The definition of $KL(F \parallel N)$ implies that words with higher probability in F and lower probability in N contribute to $KL(F \parallel N)$. Similarly, $KL(N \parallel F)$ implies that words with higher probability in F and lower probability in N contribute to $KL(N \parallel F)$. In order to compare the word distribution between fake reviews and genuine reviews, we compute the KL-divergence difference for each word, $\triangle KL^i_{word}$ as follows.

$$\triangle KL^i_{word} = KL_{word}(F_i \parallel N_i) - KL_{word}(N_i \parallel F_i) \tag{13}$$

where $KL_{word}(F_i \parallel N_i) = F(i) log\left(\frac{F(i)}{N(i)}\right)$, and similarly $KL_{word}(N_i \parallel F_i)$.

Let A denotes the set of top words contribute to $\triangle KL$, which can be further classified into sets $A^F = \{i| \triangle KL^i_{word} > 0\}$ (i.e., $\forall i \in A^F, F(i) > N(i)$) and $A^N = \{i| \triangle KL^i_{word} < 0\}$(i.e., $\forall i \in A^N, N(i) > F(i)$) where $A = A^F \cup A^N$and $A^F \cap A^N = \varnothing$. Specifically, we are more interested in words of A^F, which appear in fake reviews with much higher frequencies than in non-fake reviews. We report the top 15 words in Table 5.

These words reveal some interesting insights. First, we can identify several topics. For example, some important words are related to the food topic, such as *food*, *pizza*, and *delicious*. Reviewers use those words to express their own experiences. Fake reviewers may utilize these words to manipulate online reviews since consumers pay more attention

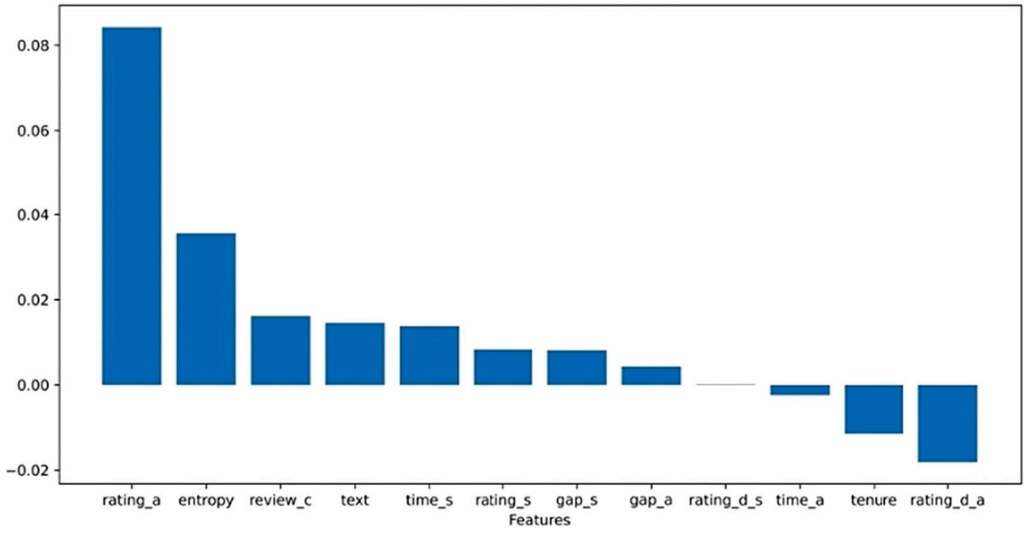


Fig. 5. Feature importance ranking of the proposed model.
Note: review_c, review count; tenure, user tenure, gap_a, review gap (avg); gap_s, review gap (std); entropy, rating entropy; rating_d_a, rating deviation (avg); rating_d_s, rating deviation (std); time_a, time of review (avg); time_s, time of review (std); rating_a, rating scores (avg); rating_s, rating scores (std).

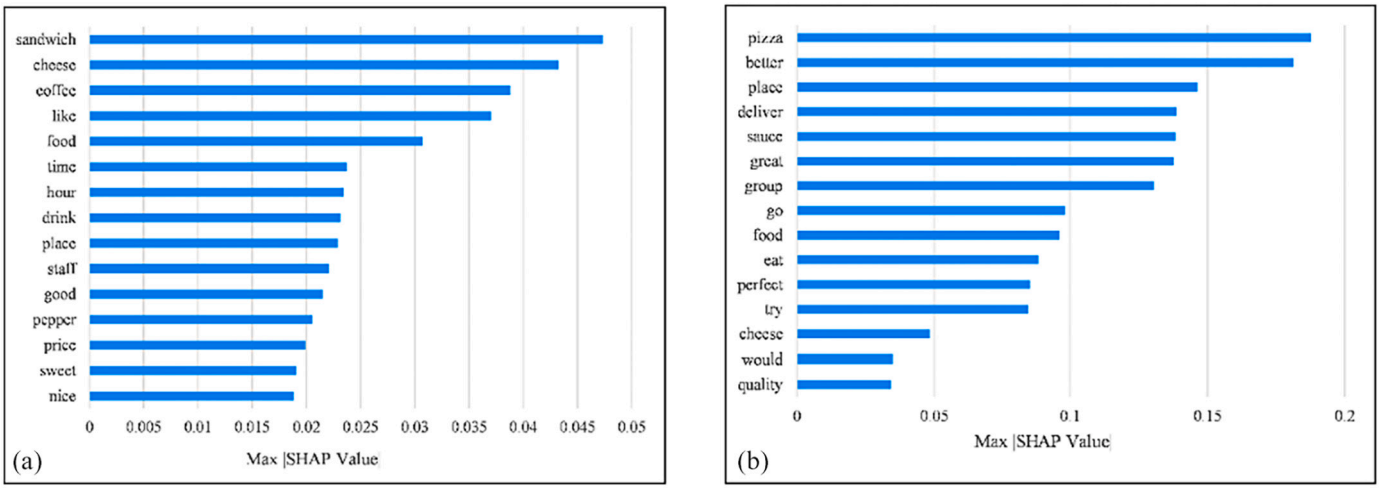


Fig. 6. (a) Top 15 words contributing to authentic reviews; (b) Top 15 words contributing to fake reviews.

to food quality. Second, we find that most of the words in Fig. 6(b) are in set A^F, which indicates that the words in A^F have great potential in improving detection performance of fake reviewers.

4.3.4. Ablation study

To further evaluate how each component of the context-aware attention mechanism contributes to the model's performance, we compared our model with its three variations in accuracy, precision, recall, f1-score, and AUC. The results are presented in Table 6.

We can obtain several crucial observations from Table 6. First, our model outperforms all its variations, demonstrating its superiority in fake reviewer detection. Second, model 3 yields the worst performance, highlighting the value of the attention mechanism in fake reviewer detection. The absence of the attention mechanism has a significant impact on model performance. Third, model 1 performs better than model 2, which indicates that CNN contributes more than BiLSTM in fake reviewer detection.

4.4. Methodology validation

We use another dataset (YelpNYC) from Rayana and Akoglu [9] to further validate our findings on the importance of advanced text analytics on fake reviewer detection. We conducted the same steps as described earlier on the YelpZIP dataset. Table 7 presents the performance of the models using all features and behavioral features only. We also present the t-test results of performance improvement obtained from behavior-sensitive feature extractor vs. other benchmarks and our model vs. other baseline methods using all features in Table 7.

The results obtained with the YelpNYC dataset are consistent with those based on the YelpZIP dataset, revealing that the proposed model with contextualized text representations from feature-based transformer adaptation produces the best performance. Besides, the *t*-test results in experiment 1 show that the behavior-sensitive feature extractor outperforms other benchmarks in terms of accuracy, recall, and AUC score. Also, the t-test results in experiment 2 indicate that incorporating contextualized text representations improved the identification performance in terms of accuracy, F1 score and AUC score.

4.5. Performance comparison with prior studies

Several studies have used online review data from Yelp to identify suspicious reviewers. To demonstrate our model's capacity to evaluate the trustworthiness of users, we compare the results with those produced by previous studies. Note that these studies and our research use the same dataset, YelpZIP.

Rayana and Akoglu [9] leverage the review metadata and textual information to identify fake reviewers, of which the AUC score is 0.683. Sandulescu and Ester [54] utilized behavior patterns and semantic similarity of review content to detect suspicious reviewers, of which the F_1-score is 0.68. Kumar et al. [8] apply the feature engineering method on behavior features to improve the classification performance of fake reviewers, of which the AUC score is 0.817. Our proposed method achieves an F_1-score of 0.7664 and an AUC score of 0.8358. Obviously, the proposed method outperforms those methods proposed by extent studies, thereby demonstrating the efficiency of our method. In addition, the proposed method automatically learns and extracts valuable features from behavior and text data, while the methods mentioned above need tedious and sometimes non-optimal feature engineering.

5. Practical and managerial implications

Organized fraud behavior on review systems, especially posting fake reviews, has seriously harmed the fairness of e-commerce platforms and users' interests. A typical scam is that companies set up a social media account or discussion group and attract users who look for free merchandise to post fake reviews in exchange for a product or a cash bonus. Deceptive endorsements between a seller and reviewer are always secretive and difficult to track. According to Fakespot[4], a tool for identifying fake reviews and counterfeits, nearly 42% of 720 million Amazon reviews evaluated in 2020 were spurious. Those fake review fraudsters and their fake reviews deeply damage the effectiveness of search rankings that rely heavily on reviews. In recent years, digital platforms have taken action to filter out fake reviews. For instance, Facebook has updated its community feedback policy to address the widespread issue of fake feedback[5]. Amazon, the largest online retailer in the world, filed legal action against fake review brokers across more than 10,000 Facebook groups in July 2022[6].

Our work provides several practical implications for improving the performance of fake reviewer detection and establishing trustworthy e-commerce platforms. First, considering that different features contribute to different levels of detection of fake reviewers (Fig. 6), it would be prudent for the managers of e-commerce platforms (e.g., Yelp) to place their focus on the key features that are most effective in identifying

[4] https://www.fakespot.com/

[5] https://www.facebook.com/business/news/keeping-reviews-authentic-trustworthy

[6] https://www.aboutamazon.com/news/policy-news-views/amazon-targets-fake-review-fraudsters-on-social-media

Table 6
Performance of our model vs. its variations on fake reviewer detection.

No.	Algorithm for BF	Algorithm for text	Acc	Precision	Recall	F1-Score	AUC
1	Conv1d	CNN + Attention	77.84%**	0.7139*	0.7255*	0.7197*	0.7778*
2	Conv1d	BiLSTM+Attention	78.48%**	0.7050**	0.6987*	0.7018*	0.7781*
3	Conv1d	CNN + BiLSTM	77.24%***	0.6884***	0.6720**	0.6801***	0.7641***
4	Our model		**85.05%**	**0.7689**	**0.7643**	**0.7664**	**0.8358**

Note: *, **, and *** denote significance level at the 0.05, 0.01, 0.001 levels, respectively. BF: Behavior feature. Numbers in Bold are the best performance in each category.

Table 7
Performance comparison of models with all features vs. behavioral features using YelpNYC.

Experiment	Algorithm for BF	Algorithm for Text	Acc	Precision	Recall	F_1-score	AUC
1 (Behavioral features only)	LR	None	71.61%**	0.6659 **	0.6628*	0.6638***	0.6888***
	RF		71.58%**	0.6652***	0.6480**	0.6563**	0.6783***
	SVM		71.55%**	0.7043	0.6629*	0.6827	0.6986**
	CART		71.34%**	0.6628**	0.6583*	0.6603**	0.6849**
	NB		72.97%**	**0.8043**	0.6409**	**0.7134**	0.7093**
	Conv1d		**74.31**%	0.6714	**0.6696**	0.6705	**0.7344**
2 (All features)	LR	LR	73.62%***	0.6728**	0.6688**	0.6702***	0.6946***
	RF	RF	73.45%***	0.6792*	0.6608**	0.6695**	0.6905***
	SVM	SVM	73.60%***	0.7038**	0.6607*	0.6813**	0.6970***
	CART	CART	72.27%***	0.6687**	0.6542**	0.6611**	0.6834***
	NB	NB	72.20%	**0.9132**	0.5420***	0.6800**	0.6824***
	Conv1d	CNN	73.69%***	0.6783**	0.6773**	0.6778**	0.7462***
		BiLSTM	74.61%**	0.6855*	0.6924**	0.6889*	0.7465*
		C-LSTM	74.68%***	0.6900*	0.6886	0.6893	0.7497**
		BERT	74.85%***	0.7056*	0.7077*	0.7066*	0.7701**
		ALBERT	74.57%***	0.6918*	0.6884*	0.6901*	0.7556**
		DistilBERT	76.08%***	0.7049	0.7109	0.7079*	0.7669**
		RoBERTa	75.94%**	0.7069	0.7092	0.7081	0.7727*
		Longformer	76.74%**	0.7151	0.7095	0.7123	0.7887*
	Our model		**80.14**%	0.7269	**0.7136**	**0.7203**	**0.7950**

Note: *, **, and *** denote significance level at the 0.05, 0.01, 0.001 levels, respectively. Numbers in Bold are the best performance in each category. BF, Behavior Features; LR, Logistic Regression; RF, Random Forest; SVM, Support Vector Machine; CART, Classification and Regression Tree; NB, Naïve Bayes; Conv1d, one-dimensional convolution; BERT, Bidirectional Encoder Representations from Transformers; DistilBERT, A Distilled Version of BERT; RoBERTa, Robustly Optimized BERT; ALBERT, A Light Version of BERT. Longformer, The Long-Document Transformer.

online fake reviewers. For example, our work shows that the average of rating scores is the most important feature for detecting fake reviewers. It might be beneficial for the Yelp management team to pay more attention to reviewers who provide more extreme ratings. Second, our study provides insights into digital traces (especially behavioral data) in fake reviewer detection and highlights the effectiveness of incorporating behavioral data in uncovering improper operating on platforms. In other words, behavior features are more effective in detecting fake reviewers than text features. Considering the importance of behavior features, managers of e-commerce platforms ought to facilitate and promote social interactions between reviewers and prospective consumers to provide rich behavior features. Third, our research shows that combining behavior and text features would be beneficial in developing models for detecting fake reviewers. It is still valuable to have text features of reviewers available, especially if behavior features are limited. The performance of text features in detecting fake reviewers could be enhanced by leveraging more advanced language models (e.g., Prompt). Our research proposes a feasible model for incorporating behavioral data and advanced language models with superior performance. Online platforms can pay more attention to the development of advanced language models in the field of natural language processing and adopt appropriate models based on the platform's capabilities.

6. Conclusions and future directions

Ensuring the fairness and justice of online reviews is a growing societal concern. Hence, there is an urgent need to address the misinformation problem caused by fake reviewers. While prevailing studies have made several attempts to detect fake reviewers, they do not fully exploit the deep learning approaches for the behavior-based and text-based features.

In this study, we propose a novel deep learning-based framework for detecting suspicious reviewers. We evaluate the proposed framework with state-of-the-art benchmarks and validate it on two real-life Yelp datasets. The empirical evaluation reveals that our proposed method can significantly increase the classification performance of fake reviewers. Our research makes two key contributions. First, we contribute to an important step toward reducing the cost of feature engineering for behavioral features. We demonstrate that leveraging feature learning for behavioral features significantly improves classification performance. Second, we contribute to fake reviewer detection literature by providing insights for extracting valuable information from texts, which provides more insights when utilizing high-level text representation to design novel artifacts for specific research inquiries.

We discuss several future research directions. First, more relevant features from the online website (e.g., user images) can be leveraged. We expect the proposed framework can incorporate more features and work efficiently with them. Second, we evaluate and validate our method on the Yelp dataset. More real datasets collected from other platforms can be further used. Third, our method can be generated to identify suspicious users from other online platforms (e.g., anomaly users of online health forums) as a promising future research direction.

CRediT authorship contribution statement

Dong Zhang: Writing – original draft, Investigation, Software. **Wenwen Li:** Conceptualization. **Baozhuang Niu:** Conceptualization, Supervision, Validation, Writing – review & editing. **Chong Wu:**

Conceptualization.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

The authors are grateful to the editor and reviewers for their helpful comments. This work was supported by the National Natural Science Foundation of China (72201105, 72125006).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dss.2022.113911.

References

[1] M.L. Jensen, J.M. Averbeck, Z. Zhang, K.B. Wright, Credibility of anonymous online product reviews: a language expectancy perspective, J. Manag. Inf. Syst. 30 (1) (2013) 293–323.
[2] W. Zhou, W. Duan, Do professional Reviews affect online user choices through user reviews? An empirical study, J. Manag. Inf. Syst. 33 (1) (2016) 202–228.
[3] M.G. Lozano, J. Brynielsson, U. Franke, et al., Veracity assessment of online data, Decis. Support. Syst. 129 (2020), 113132.
[4] D. Zhang, L. Zhou, J.L. Kehoe, I.Y. Kilic, What online reviewer behaviors really matter? Effects of verbal and nonverbal behaviors on detection of fake online reviews, J. Manag. Inf. Syst. 33 (2) (2016) 456–481.
[5] Y. Wu, E.W.T. Ngai, P. Wu, C. Wu, Fake online reviews: literature review, synthesis, and directions for future research, Decis. Support. Syst. 132 (2020), 113280.
[6] X. Ma, L. Khansa, Y. Deng, S.S. Kim, Impact of prior reviews on the subsequent review process in reputation systems, J. Manag. Inf. Syst. 30 (3) (2013) 279–310.
[7] M. Ott, C. Cardie, J. Hancock, Estimating the prevalence of deception in online review communities, in: Proceedings of the 21st international conference on World Wide Web, 2012. Lyon, France.
[8] N. Kumar, D. Venugopal, L.F. Qiu, S. Kumar, Detecting review manipulation on online platforms with hierarchical supervised learning, J. Manag. Inf. Syst. 35 (1) (2018) 350–380.
[9] S. Rayana, L. Akoglu, Collective opinion spam detection: bridging review networks and metadata, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015. Sydney, NSW, Australia.
[10] A. Mukherjee, A. Kumar, B. Liu, et al., Spotting opinion spammers using behavioral footprints, in: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, 2013. Chicago, Illinois, USA.
[11] N.A. Patel, R. Patel, A survey on fake review detection using machine learning techniques, in: 2018 4th International Conference on Computing Communication and Automation (ICCCA), Greater Noida, India, 2018, pp. 1–6.
[12] L. Li, B. Qin, W. Ren, T. Liu, Document representation and feature combination for deceptive spam review detection, Neurocomputing. 254 (2017) 33–41.
[13] B. Manaskasemsak, J. Tantisuwankul, A. Rungsawang, Fake review and reviewer detection through behavioral graph partitioning integrating deep neural network, Neural Comput. & Applic. (2021) 1–14.
[14] G. Papadakis, G. Giannakopoulos, G. Paliouras, Graph vs. bag representation models for the topic classification of web documents, World Wide Web. 19 (5) (2016) 887–920.
[15] M. Diale, T. Celik, C. Van Der Walt, Unsupervised feature learning for spam email filtering, Comput. Electr. Eng. 74 (2019) 89–104.
[16] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[17] I. Beltagy, M.E. Peters, A. Cohan, Longformer: The Long-Document Transformer, arXiv:2004.05150, https://ui.adsabs.harvard.edu/abs/2020arXiv200405150B, 2020.
[18] M. Samadi, M. Mousavian, S. Momtazi, Deep contextualized text representation and learning for fake news detection, Inf. Process. Manag. 58 (6) (2021), 102723.
[19] N. Kumar, D. Venugopal, L.F. Qiu, S. Kumar, Detecting anomalous online reviewers: an unsupervised approach using mixture models, J. Manag. Inf. Syst. 36 (4) (2019) 1313–1346.
[20] H. Li, G. Fei, S. Wang, et al., Bimodal distribution and co-bursting in review spam detection, in: Proceedings of the 26th International Conference on World Wide Web, 2017. Perth, Australia.
[21] G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, R. Ghosh, Exploiting burstiness in reviews for review spammer detection, in: International Conference on Weblogs and Social Media, Cambridge, Massachusetts, USA, 2013, pp. 175–184.
[22] A. Mukherjee, B. Liu, N. Glance, Spotting fake reviewer groups in consumer reviews, in: Proceedings of the 21st international conference on World Wide Web, 2012. Lyon, France.
[23] P. Gupta, S. Gandhi, B.R. Chakravarthi, Leveraging Transfer learning techniques-BERT, RoBERTa, ALBERT and DistilBERT for Fake Review Detection. Forum for Information Retrieval Evaluation, 2021. Virtual Event, India.
[24] R. Mohawesh, S. Xu, S.N. Tran, et al., Fake reviews detection: a survey, IEEE Access. 9 (2021) 65771–65802.
[25] W. Liu, W. Jing, Y. Li, Incorporating feature representation into BiLSTM for deceptive review detection, Computing. 102 (3) (2020) 701–715.
[26] A. Rastogi, M. Mehrotra, S.S. Ali, Effective opinion spam detection: a study on review metadata versus content, J. Data Inform. Sci. 5 (2) (2020) 76–110.
[27] S. Noekhah, Salim Nb, N.H. Zakaria, Opinion spam detection: using multi-iterative graph-based model, Inf. Process. Manag. 57 (1) (2020), 102140.
[28] H. Aghakhani, A. Machiry, S. Nilizadeh, C. Kruegel, G. Vigna, Detecting deceptive reviews using generative adversarial networks, in: 2018 IEEE Security and Privacy Workshops (SPW), 2018, pp. 89–95.
[29] C.-C. Wang, M.-Y. Day, C.-C. Chen, J.-W. Liou, 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, 2018. Hong Kong, Hong Kong.
[30] X. Wang, K. Liu, J. Zhao, Detecting deceptive review spam via attention-based neural networks, in: Natural Language Processing and Chinese Computing, Cham, 2018, pp. 866–876.
[31] Y. Ren, Y. Zhang, Deceptive opinion spam detection using neural network, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 2016, pp. 140–150.
[32] F. Li, M. Huang, Y. Yang, X. Zhu, Learning to identify review spam, in: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, 2011. Barcelona, Catalonia, Spain.
[33] Y. Shang, M. Liu, T. Zhao, J. Zhou, T-Bert: A Spam Review Detection Model Combining Group Intelligence and Personalized Sentiment Information, Cham, 2021, pp. 409–421.
[34] D. Refaeli, P. Hajek, Detecting fake online reviews using fine-tuned BERT, in: 2021 5th International Conference on E-Business and Internet, 2021. Singapore, Singapore.
[35] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv: 181004805, 2018.
[36] J. Ye, L. Akoglu, Discovering Opinion Spammer Groups by Network Footprints, Cham, 2015, pp. 267–282.
[37] J.G. Proudfoot, J.L. Jenkins, J.K. Burgoon, J.F. Nunamaker, More than meets the eye: how oculometric behaviors evolve over the course of automated deception detection interactions, J. Manag. Inf. Syst. 33 (2) (2016) 332–360.
[38] G. Shan, L. Zhou, D. Zhang, From conflicts and confusion to doubts: examining review inconsistency for fake review detection, Decis. Support. Syst. 144 (2021), 113513.
[39] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep learning for computer vision: a brief review, Comput. Intell. Neurosci. 2018 (2018).
[40] D.W. Otter, J.R. Medina, J.K. Kalita, A survey of the usages of deep learning for natural language processing, IEEE Trans. Neural Netw. Learn. Syst. 32 (2) (2021) 604–624.
[41] Y. Liu, Y.-F.B. Wu, FNED: a deep network for fake news early detection on social media, ACM Trans. Inf. Syst. 38 (3) (2020) (Article 25).
[42] S. Banerjee, A.Y.K. Chua, A study of manipulative and authentic negative reviews, in: Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication, Siem Reap, Cambodia, 2014. Article 76.
[43] M. Ott, Y. Choi, C. Cardie, J.T. Hancock, Finding deceptive opinion spam by any stretch of the imagination, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, 2011. Portland, Oregon.
[44] A. Vaswani, N. Shazeer, N. Parmar, et al., Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008. Long Beach, California, USA.
[45] Y. Liu, M. Ott, N. Goyal, et al., Roberta: A Robustly Optimized Bert Pretraining Approach, arXiv Preprint arXiv:190711692, 2019.
[46] R.K. Kaliyar, A. Goswami, P. Narang, FakeBERT: fake news detection in social media with a BERT-based deep learning approach, Multimed. Tools Appl. 80 (8) (2021) 11765–11788.
[47] M.E. Basiri, S. Nemati, M. Abdar, E. Cambria, U.R. Acharya, ABCDM: an attention-based bidirectional CNN-RNN deep model for sentiment analysis, Futur. Gener. Comput. Syst. 115 (2021) 279–294.
[48] T. Islam, D. Goldwasser, Analysis of twitter users' lifestyle choices using joint embedding model, Proceedings of the International AAAI Conference on Web and Social Media. 15 (1) (2021) 242–253.
[49] A. Mukherjee, V. Venkataraman, B. Liu, N. Glance, What yelp fake review filter might be doing?, in: Proceedings of the 7th International Conference on Weblogs and Social Media, ICWSM 2013, Cambridge, Massachusetts, USA, 2013, pp. 409–418.
[50] G. Satia Budhi, R. Chiong, Z. Wang, S. Dhakal, Using a hybrid content-based and behaviour-based featuring approach in a parallel environment to detect fake reviews, Electron. Commer. Res. Appl. 47 (2021), 101048.

[51] A. Altmann, L. Toloşi, O. Sander, T. Lengauer, Permutation importance: a corrected feature importance measure, Bioinformatics. 26 (10) (2010) 1340–1347.
[52] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Proceedings of the 31st international conference on neural information processing systems, 2017, pp. 4768–4777.
[53] A. Mukherjee, V. Venkataraman, B. Liu, N.S. Glance, Fake Review Detection : Classification and Analysis of Real and Pseudo Reviews, Department of Computer Science, University of Illinois at Chicago, 2013.
[54] V. Sandulescu, M. Ester, Detecting singleton review spammers using semantic similarity, in: Proceedings of the 24th International Conference on World Wide Web, 2015. Florence, Italy.

Dong Zhang is currently working as a postdoc in the School of Business Administration, South China University of Technology, Guangzhou, China. His papers have been published at journals such as Journal of the Operational Research Society and Kybernetes. His research interests include data mining methods and online reviews.

Wenwen Li is currently working as an Assistant Professor in School of Management, Fudan University, Shanghai, China. Her papers once appeared at journals such as MIS Quarterly. Her research interests include online auction and bayesian updating models.

Baozhuang Niu is currently a Full Professor with the South China University of Technology, Guangzhou, China. He has authored/coauthored 11 top journal papers including Manufacturing & Service Operations Management (two papers), Production and Operations Management (six papers), and TRB (three papers), among other peer-review journal papers. His research interests include supply chain operations.

Chong Wu is current a Full Professor with School of Economics and Management, Harbin Institute of Technology, Harbin, China. His papers have appeared in journals such as the IEEE Transactions on Knowledge and Data Engineering, Journal of the Association for Information Science and Technology, Information Sciences, and Knowledge-based Systems.