Electronic Commerce Research (2023) 23:2737–2757
https://doi.org/10.1007/s10660-022-09560-w

Effectiveness of Fine-tuned BERT Model in Classification of Helpful and Unhelpful Online Customer Reviews

Muhammad Bilal[1] · **Abdulwahab Ali Almazroi**[2]

Accepted: 5 April 2022 / Published online: 29 April 2022

Abstract

The problem of information overload in online review platforms has seriously hampered many customers' ability to evaluate the quality of products or businesses when making purchasing decisions. A large body of literature exists that attempts to predict the helpfulness of online customer reviews and has reported contradictory findings on the effectiveness of various approaches. Moreover, many existing solutions use traditional machine learning techniques and handcrafted features, limiting generalization. Therefore, this study aims to propose a generalized approach by fine-tuning the BERT (Bidirectional Encoder Representations from Transformers) base model. The performance of BERT-based classifiers is then compared with that of bag-of-words approaches to determine the effectiveness of BERT-based classifiers. The evaluations performed using Yelp shopping reviews show that fine-tuned BERT-based classifiers outperform bag-of-words approaches in classifying helpful and unhelpful reviews. In addition, it is found that the sequence length of the BERT-based classifier has a significant impact on classification performance.

Keywords Electronic Word-Of-Mouth · Online Reviews · Review Helpfulness · Sequence Classification · Bag-of-Words · Deep Learning

✉ Muhammad Bilal
bilal.m@nu.edu.pk

Abdulwahab Ali Almazroi
aalmazroi@uj.edu.sa

[1] Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Chiniot-Faisalabad Campus, 35400 Chiniot, Pakistan

[2] College of Computing and Information Technology at Khulais, Department of Information Technology, University of Jeddah, Jeddah 21959, Saudi Arabia

1 Introduction

Recent advancements in web technologies, as well as the COVID-19 pandemic, have accelerated digital transformation and increased use of social media and e-commerce platforms, resulting in massive amounts of User-Generated Content (UGC) [1, 2]. Electronic word-of-mouth (eWOM), such as online customer reviews, is a popular and growing source of UGC hosted by a variety of review websites such as Amazon, Yelp, and TripAdvisor, etc. [3]. Prospective customers use online customer reviews to assess the quality of a product, business, or service before making a purchase decision. According to reports, online customer reviews have a significant impact on the purchasing decisions of many potential customers [4, 5]. Furthermore, online customer reviews assist businesses in analyzing and managing customer needs, which leads to increased customer satisfaction [6]. Millions of reviews have been posted on review websites such as Yelp, which has 224 million reviews [7]. The number of online customer reviews is growing at an exponential rate, presenting several opportunities and challenges for both businesses and customers [8, 9]. The extremely inconsistent quality and the sheer volume of online customer reviews have resulted in a problem of information overload, making it difficult for potential customers to find helpful reviews due to limited cognitive abilities [10, 11].

Review platforms have introduced and implemented helpful votes that tap into crowd wisdom to combat the problem of information overload [12]. The most important aspect of online reviews is their helpfulness, which reflects the subjective nature and perceived quality of online reviews by readers [13]. Readers believe that reviews with a higher percentage of helpful votes are more credible than reviews with no helpful votes [14]. However, a significant portion of online customer reviews did not receive helpful votes, particularly for businesses or products that received a high volume of reviews and the most recent reviews, which did not have enough time to receive helpful votes [15, 16]. As a result, reviews with helpful votes are scarce, making it difficult to assess the quality of a product or business and make purchasing decisions [17, 18]. Therefore, researchers have proposed several statistical and machine learning solutions for identifying important features and predicting the helpfulness of online customer reviews. In the literature, there are three major types of features for predicting review helpfulness: review content, business or product features, and reviewer features. However, in comparison to other features, more emphasis is placed on the use of review-related features. Existing studies have identified review age, length, rating, readability, polarity, and subjectivity as important features for predicting review helpfulness [19, 20].

The existing literature defines review helpfulness in two ways. First, it is defined as the ratio of helpful votes to total votes received by a review. Second, it is defined as the total number of helpful votes received by a review. However, the solutions presented for the ratio-based definition of review helpfulness are no longer valid because review platforms have removed information about total votes. Furthermore, researchers have focused on both regression and classification tasks for predicting review helpfulness and classifying helpful and unhelpful reviews [19]. Despite the growing interest of researchers in predicting the helpfulness of online customer reviews, the majority of the features used are handcrafted. According to recent surveys, the existing literature on review helpfulness prediction is “disorganized” owing to contradictory findings on the importance of features. [19, 21]. The majority of existing solutions for review helpfulness are based on traditional machine learning algorithms such as Linear Regression (LNR), Random Forest (RF), Support Vector Machine (SVM). Whereas only a few studies have attempted to use cutting-edge approaches, such as Deep Learning (DL), to predict review helpfulness [19]. Furthermore, many studies reported contradictory and conflicting results on the effectiveness of various approaches for predicting review helpfulness. Hence, the effectiveness of state-of-the-art approaches for automatically extracting features and predicting the helpfulness of online reviews needed to be investigated further.

BERT is a cutting-edge deep learning technique based on transformers for Natural Language Understanding (NLU) tasks [22]. Since the release of BERT in 2018, only two studies [23, 24] have been conducted to investigate its effectiveness in predicting the helpfulness of online reviews. [23] defined helpfulness as a count variable and used BERT to classify helpful and unhelpful reviews. [24], on the other hand, looked at review helpfulness from a regression perspective and defined it as a ratio of helpful to total votes. However, both previous studies were limited by the use of small datasets of Amazon reviews. In addition, both studies reported contradictory results on the effectiveness of BERT for predicting review helpfulness. Furthermore, according to a survey on review helpfulness prediction [19], the dataset and definition of helpfulness used by [24] are depreciated. This study attempts to fill this research gap and provide timely insights by fine-tuning BERT for the classification of helpful and unhelpful reviews. Contextualized token embeddings are generated using the BERT tokenizer, which eliminates the need for handcrafted features. The performance of a fine-tuned BERT for review classification is evaluated through training and testing with reviews extracted from the Yelp Open Dataset. The performance of BERT-based classifiers is compared to that of bag-of-words approaches such as k-nearest neighbor (k-NN), Naïve Bayes (NB), and SVM to determine the effectiveness of BERT-based classifiers for classifications of reviews. The length of online reviews can vary significantly, but both existing studies have used only a single value of sequence length to predict review helpfulness. As per our knowledge, no study has yet explored the impact of using different sequence lengths in BERT on review helpfulness. Therefore, this study also considers how different sequence lengths affect the performance of a fine-tuned BERT for the task of sequence classification.

The rest of the paper is divided into sections. Section 2 provides a brief overview of the existing literature. The research methodology used in this study is described in Section 3. Section 4 presents and discusses the results. The implications of this research are discussed in Section 5. Finally, Section 6 concludes this study.

2 Literature Review

Researchers have paid increasing attention to the task of predicting helpfulness over the last decade. Several solutions based on various features and machine learning algorithms have been proposed and evaluated using real-time datasets from Amazon, Yelp, Tripadvisor, and others. This section provides an in-depth overview of the existing literature on review helpfulness prediction. Kim et al. [15] set the basis for automatic prediction of review helpfulness in 2006. The structural, syntactic, semantic, and meta-data features extracted from Amazon product reviews were used to develop a regression model with a 0.66 correlation. Significant features that have been reported include review length, uni-grams, and product rating. According to Liu et al. [25], the helpfulness of online reviews is determined by the reviewer's experience, writing style, and timeliness of the review. The results of a non-linear model developed using these features and evaluated using IMDb movie reviews demonstrated the effectiveness of the proposed approach. Lee and Choeh [26] predicted the helpfulness of online reviews by combining product, review metadata, and review textual features. DNN was found to be more accurate than LNR at predicting review helpfulness.

Krishnamoorthy [27] proposed a model for predicting the helpfulness of reviews based on novel linguistic features extracted from the review text. In addition to linguistic features, review metadata, subjectivity, and readability were also used. Experiments with Amazon product reviews revealed that the proposed linguistic features were effective in predicting the helpfulness of reviews for experience products. Hu and Chen [28] studied the effect of review, sentiment, visibility, and reviewer-related features on the helpfulness of online reviews. It was determined that the visibility features are strongly related to review helpfulness, and that using these features significantly improved performance. Singh et al. [29] proposed a method for predicting the helpfulness of reviews based on various review-related features. Polarity, subjectivity, and entropy were found to be the most important textual features, while the length of review, stop words, and wrong words were found to be less important. Hu et al. [30] developed three user-controllable filters and applied them to predict the helpfulness of TripAdvisor reviews. It was observed that the most important features for predicting helpfulness are review rating and review length. It was also seen that RF outperformed the other machine learning algorithms.

According to Chen et al. [31], existing approaches for predicting review helpfulness require labeled reviews for each category. However, it did not reflect the real-world scenario where some domains did not have sufficient reviews with helpful votes. However, it did not reflect the real-world situation in which some domains lacked sufficient reviews with helpful votes. Therefore, a CNN model was created using word and character representations to use a transfer learning approach for cross-domain review helpfulness prediction. According to Zhang and Lin [32], existing literature is only focused on the use of reviews written in English, and no study has attempted to predict the helpfulness of non-English reviews. Therefore, a multilingual approach was proposed in which non-English

Yelp reviews are converted to English and used to predict review helpfulness. Akbarabadi and Hosseini [33] investigated the role of title features in predicting the helpfulness of reviews. The results revealed that title features are not an important determinant of review helpfulness when compared to other features.

Ma et al. [34] argued that existing solutions for predicting review helpfulness did not take user-supplied photos into account. The impact of photos and other features was studied using TripAdvisor and Yelp reviews. The experiment results revealed that deep learning algorithms outperform other machine learning algorithms. It was discovered that using a hybrid combination of features yields the best results. Saumya et al. [35] predicted review helpfulness using data from consumer question answers, as well as review and product features. The predicted helpfulness score was then used to rank reviews. Lee et al. [36] investigated the effect of review quality, sentiment, and reviewer-related characteristics on the helpfulness of online reviews. When the performance of different machine learning algorithms in classifying helpful and unhelpful reviews was compared, it was discovered that RF produced the best results. The evaluations conducted using the TripAdvisor review dataset revealed that reviewer features are more important than review quality and sentiment features.

Sun et al. [37] proposed a review informativeness measurement and investigated its impact on review helpfulness prediction for search and experience products. It was seen that informativeness of review significantly improves prediction accuracy compared to review length. Olatunji et al. [38] also proposed a DNN-based context-aware review helpfulness prediction model and validated its performance using a human-annotated Amazon dataset. Du et al. [39] argued that existing review helpfulness prediction approaches lack broad generalization due to platform-specific and hand-crafted features. To address this, thirty features from the literature were identified and classified into five categories. The experiments were divided into three categories: using single features, using category-based features, and using all features. In comparison to other features, semantic features were found to play a significant role in review helpfulness prediction. Reviews can be recommended based on the helpfulness votes. However, most of the reviews received no helpful votes, limiting the task of making recommendations. Ge et al. [40] proposed a review recommendation technique based on the helpfulness scores predicted by a model trained on reviews with helpful votes.

Chen et al. [41] also proposed a gated CNN based on the transfer learning approach for cross-domain review helpfulness prediction. Luo and Xu [42] used Latent Dirichlet Allocation (LDA) to extract aspects from online restaurant reviews. The extracted aspects are then used to develop a classification model using SVM with Fuzzy Domain Ontology (FDO). Experiment results using the Yelp dataset demonstrated that SWM with FDO outperforms traditional classification algorithms. Saumya et al. [43] proposed a CNN-based model for predicting review helpfulness using review representation learning. A pre-trained model was used to convert review text into a low-dimensional vector for automatic feature extraction. The proposed method, which is based on textual features such as trigrams, fourgrams, and fivegrams, outperforms previous studies that used hand-crafted features. According to Fan et al. [44], only extracting and linguistic features for review text did not fully reflect the helpfulness. The helpfulness of online reviews should be aware of product metadata. A proposed DNN model takes metadata features and review text directly to perform product-aware review helpfulness prediction. The experiments done using Amazon and Yelp reviews produced state-of-the-art results.

Kong et al. [45] emphasized that existing approaches for predicting review helpfulness required hand-crafted features to make predictions. Hence, a novel hybrid approach based on the combination of Convolutional Neural Network (CNN) and Translating Embeddings (TransE) was proposed to eliminate the need for hand-crafted features. Furthermore, the proposed method allows for the inclusion of hand-crafted features, which further improve prediction performance. Review depth is defined in the literature as the number of words in a review text. Son et al. [46] defined review breath based on topics in a review text that reflect the information contained in a review. The experiment results showed that using review breath in addition to existing features significantly improved performance. Malik and Hussain [47] introduced several influential review, product, and reviewer features to predict review helpfulness. It was discovered that using only newly introduced review and reviewer features yields the best prediction performance. The recency and length of activity have been identified as important reviewer-related features.

Wu and Chen [23] argued that the helpfulness of reviews is not constant and changes over time. Therefore, information for helpfulness votes for the same Amazon product reviews is gathered over eight weeks. BERT was used to classify reviews that had zero and non-zero helpful votes. The results showed that for a similar domain F1 score of 0.732 was achieved compared to an F1 score of 0.550 for cross-domain. Xu et al. [24] also tested the accuracy of the BERT in predicting the helpfulness of Amazon product reviews. The findings were intriguing because BERT did not achieve the best performance when compared to Support Vector Regression (SVR) and RF for various datasets. However, the BERT had the best overall average performance across all datasets. Moreover, the optimization of hyperparameters was not done due to limited resources. Du et al. [48] argued that most of the exiting models are using star rating and review text for review helpfulness prediction in a way that does not fully exploit review star ratings. To model review helpfulness, a deep neural architecture was proposed that uses text-rating interaction. The proposed approach outperformed existing state-of-the-art techniques in terms of prediction accuracy in experiments using the Amazon dataset.

Namvar [49] proposed a novel review helpfulness prediction technique that cluster reviews based on reviewer and temporal features. Afterwards, the review helpfulness prediction is performed for each cluster based on review features. The evaluations performed using Amazon dataset showed that the proposed approach showed better performance compared to existing approaches. Malik [50] investigated the effect of review, reviewer, and product features on review helpfulness. Several models for predicting review helpfulness were developed and tested using Amazon product reviews. The results showed that the Deep Neural Network (DNN) prediction model performed better. The key features that influence review helpfulness were identified as polarity of review title, polarity of review text, and concise similarity of product title and review text.

Bilal et al. [20] recently profiled reviewer business choice and rating behavior and investigated its impact on the helpfulness of Yelp business reviews. Bilal et al. [51] also proposed reviewer network strength features and used them to predict review helpfulness. Forty features related to review, business, and reviewer were used to model review helpfulness using various machine learning algorithms. According to the results, Bagging Gradient-Boosted Trees and a hybrid combination of features outperformed all others. In addition to review features, the nine key features included business and reviewer features. The results were impressive, but the use of hand-crafted features was a drawback. Mauro et al. [52] proposed a novel method for predicting the helpfulness of online customer reviews that incorporates the rating, length, and polarity deviations of reviews written by a single user or business. It was discovered that user-based rating and review length deviations have a significant impact on review helpfulness.

Du et al. [53] argued that because online reviews are presented in a sequence with other nearby reviews, review helpfulness is rarely perceived independently. Therefore, several schemes for incorporating contextual cues from neighboring reviews and analyzing the impact of neighboring reviews on perceived helpfulness were proposed. The findings confirmed that perceived helpfulness of reviews is influenced by its surrounding reviews. Lee et al. [54] proposed several prediction models for the helpfulness of Yelp business reviews using a variety of machine learning techniques. According to the findings, extreme gradient boosting outperformed LNR, RF, and SVR in prediction performance. A recent study [55] attempted to investigate the effect of argumentation and review length on helpfulness using Amazon product reviews. Longer reviews with a high rate of change in argumentation (positive and negative) were found to be less helpful. Furthermore, Kashyap et al. [56] discovered that the length of the review has no positive impact on its helpfulness based on data collected from a focused group. Olmedilla et al. [57] used 1D CNN, which uses cluster analysis to automatically classify helpful and helpful reviews. The proposed method achieved 66% accuracy, highlighting the importance of review content and context in the classification of helpful and unhelpful reviews.

The literature reviewed shows that existing approaches for predicting the helpfulness of online customer reviews have several limitations and challenges. A wide range of features has been used in the literature to predict the helpfulness of the reviews. However, the majority of the features utilized by existing studies are handcrafted or require specialized pre-processing, limiting the generalizability of proposed solutions. The findings of exiting studies on the importance of the various features are inconsistent and contradictory which also require further investigation. It is also clear that only a few studies have attempted to predict the helpfulness of reviews using cutting-edge approaches such as BERT. Moreover, the predictive performance of existing solutions to predict the helpfulness of online customer reviews has varied considerably and reported contradictory results on the effectiveness of various approaches. Therefore, solutions based on the cutting-edge approaches for review helpfulness prediction are required that ensure generalization and consistent performance by overcoming limitations associated with existing solutions.

3 Research Methodology

This section describes data collection, data labeling, train and test datasets, fine-tuning BERT, training bag-of-words approaches, and evaluating classifier performance for identifying helpful and unhelpful reviews. Fig. 1 illustrates the flow of steps in research methodology.

3.1 Data Collection and Labeling

This study uses Yelp reviews to classify helpful and unhelpful reviews. Yelp is a popular crowd-sourced business review platform that is introduced in 2004 [7]. It allows users to review and rate businesses in a variety of categories such as restaurants, shopping, beauty and fitness, and so on. In early 2019, Yelp released the Yelp Open Dataset, which contains 8.6 million reviews spanning approximately fourteen years, from October 12, 2004, to November 14, 2018. [58]. Bilal et al. [51] selected and created a dataset of 48,442 shopping reviews from the Yelp Open dataset and used it for the classification of helpful and unhelpful reviews. Reviews with no helpful votes are considered unhelpful, whereas reviews with four or more helpful votes are considered helpful. Reviews with one, two, and three votes were discarded by [51] to avoid voting bias and class overlap, in accordance with existing literature [40]. The dataset used by Bilal et al. [51] has 23127 “helpful” and 25315 “unhelpful” reviews. The stratified sampling technique is used in this study to select 10,000 reviews from the dataset used by Bilal et al. [51]. The resulting dataset contains 5000 “helpful” and “5000” unhelpful reviews. More information on the description of the dataset used in this study is provided in the following section.

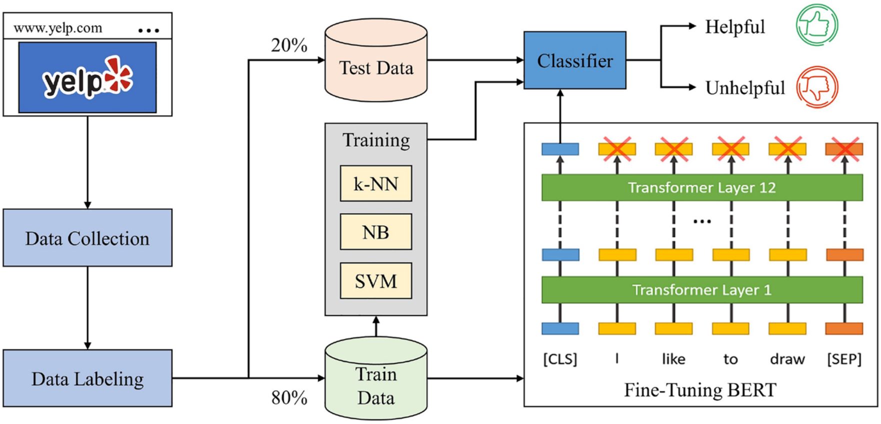


Fig. 1 Flow of steps in research methodology

3.2 Dataset

A dataset of 10000 Yelp shopping reviews with an equal representation of helpful and unhelpful reviews is used in this study. The dataset is divided into train and test datasets using a stratified sampling technique. 80% (8000) of the reviews are kept for training purposes, while the remaining 20% (2000) are used for testing purposes. 10% (800) of the reviews from the training dataset are used for validation during each training cycle in fine-tuning BERT. Table 1 contains a detailed description of the dataset used in this study. Table 1 also shows the Maximum (Max), Minimum (Min), and Average (Avg) length of reviews based on the number of words in the train, test, and overall datasets. There is a noticeable difference in the average length of helpful and unhelpful reviews. Furthermore, the Avg length for helpful and unhelpful reviews is nearly identical across all datasets.

3.3 Fine-Tuning BERT

BERT has demonstrated state-of-the-art performance on a wide range of NLU tasks, including text classification and question answering. Furthermore, the benefits of using BERT include faster development, fewer data requirements, and improved results [22]. The steps involved in fine-tuning BERT are as follows.

Input Formatting for BERT: The data must be transformed into a specific format for BERT to be trained on it. To accomplish this, the text of the reviews is first tokenized with the uncased BERT tokenizer. Following the creation of word tokens, a special token [CLS] is appended at the start and a special token [SEP] is appended at the end. The [CLS] token must be added at the beginning of classification tasks. The generated tokens are then mapped to their indexes in the tokenizer vocabulary. The sequence length (maximum of 512) is then chosen, and all reviews are truncated or padded to a single, fixed length. In this study, six different sequence lengths of 64, 128, 256, 320, 384, and 512 are used for experimentation. Finally, attention masks are created to distinguish between real and padded tokens. These steps are demonstrated in the following example with a sequence length of 64.

Table 1 Description of datasets

Dataset	Size	Class	Max Length	Min Length	Avg Length
Train	4000	Helpful (1)	987	3	186.03
	4000	Unhelpful (0)	825	2	98.89
	8000	both	987	2	142.46
Test	1000	Helpful (1)	1012	10	187.25
	1000	Unhelpful (0)	631	2	101.09
	2000	both	1012	2	144.17
Overall	5000	Helpful (1)	1012	3	186.27
	5000	Unhelpful (0)	825	2	99.33
	10000	both	1012	2	142.8

Input Text: ’i ve been buying flowers from this shop for nearly a decade whenever i send my mother flowers i call dibella and they always get it right’

Tokenized: [’i’, ’ve’, ’been’, ’buying’, ’flowers’, ’from’, ’this’, ’shop’, ’for’, ’nearly’, ’a’, ’decade’, ’whenever’, ’i’, ’send’, ’my’, ’mother’, ’flowers’, ’i’, ’call’, ’di’, ’##bella’, ’and’, ’they’, ’always’, ’get’, ’it’, ’right’]

Special Tokens: [’[CLS]’, ’i’, ’ve’, ’been’, ’buying’, ’flowers’, ’from’, ’this’, ’shop’, ’for’, ’nearly’, ’a’, ’decade’, ’whenever’, ’i’, ’send’, ’my’, ’mother’, ’flowers’, ’i’, ’call’, ’di’, ’##bella’, ’and’, ’they’, ’always’, ’get’, ’it’, ’right’, ’[SEP]’]

Token IDs: [101, 1045, 2310, 2042, 9343, 4870, 2013, 2023, 4497, 2005, 3053, 1037, 5476, 7188, 1045, 4604, 2026, 2388, 4870, 1045, 2655, 4487, 21700, 1998, 2027, 2467, 2131, 2009, 2157, 102]

Added Attention Masks:[101, 1045, 2310, 2042, 9343, 4870, 2013, 2023, 4497, 2005,3053, 1037, 5476, 7188, 1045, 4604, 2026, 2388, 4870, 1045, 2655, 4487, 21700, 1998, 2027, 2467, 2131, 2009, 2157, 102, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

This example above also shows the handling of tokens that are not present in the tokenizer vocabulary. Like in the original text “dibella” is split into two tokens “di” and “##bella”. The token ids after adding attention masks are then used to fine-tune the BERT base model for classification.

Training BERT Classifier: The token ids, attention masks, and labels of the training dataset are combined into TensorDataset. The samples in TensorDataset are then randomly split into 90-10 train-validation datasets. According to the train-validation split, 7200 samples are used for training and 800 samples are used for validation. The BERT base model has 12 transformer layers, 12 attention heads, 768 hidden layers, and a maximum sequence length of 512. Google Colab is used in this study to fine-tune BERT for the classification task. Furthermore, Google Colab assigns 1 free GPU (Tesla T4 or Tesla k80) based on availability. This study uses “BertForSequenceClassification”, a simple BERT model with a single layer added on top for classification. BERT authors recommend the hyperparameters used for fine-tuning [22]. For training and validation of classifiers developed with sequence lengths of 64, 128, 256, and 320, a batch size of 32 is used. For a sequence length of 384, a batch size of 16 is used. A batch size of 8 is used for a sequence length of 512 to avoid memory issues. For fine-tuning BERT, the authors recommend batch sizes of 16 and 32 [22]. The hyperparameters used for fine-tuning the BERT base model are given in Table 2. The optimizer is responsible for updating parameters for each batch in an epoch. The output of each training cycle is evaluated by calculating validation loss and accuracy using a validation split. Six different classifiers are fine-tuned in this study based on different sequence lengths.

3.4 Bag-of-Words Based Approaches

The effectiveness of the fine-tuned BERT base model cannot be determined unless it is compared to non-BERT models. Hence, three text classifiers, k-NN, NB, and SVM, are trained for the classification of helpful and unhelpful reviews in this study. The bag-of-words model and Term Frequency-Inverse Document

Table 2 Hyperparameters for fine-tuning BERT

Sequence Length	Batch size	Epochs	Learning rate	Epsilon (eps)
64	32	4	2e-5	1e-8
128	32			
256	32			
320	32			
384	16			
512	8			

Frequency (TF-IDF) are used to generate textual features from review text, which are then used to train k-NN, NB, and SVM. Several pre-processing steps including case transformation, tokenization, filter stop words, and stemming are performed to generate unigrams. After that, word vectors based on TF-IDF are generated, and features with missing values are removed. A total of 18982 features are generated, with less than 1% of them being pruned, resulting in a total of 880 features that are used further. The hyperparameters optimized using grid search for training k-NN, NB, and SVM are shown in Table 3.

3.5 Performance Evaluation

In this study, the problem of predicting review helpfulness is treated as a binary classification task, with "helpful" reviews labeled as positive or true (1) and "unhelpful" reviews labeled as negative or false (0). The performance of the BERT and bag-of-words based classifiers is evaluated using a test dataset of 2000 reviews. The test dataset is also passed through the input formatting steps to be converted into the BERT required format. Moreover, for testing k-NN, NB, and SVM, a number of pre-processing steps are also performed to generate unigrams. Afterward, word vectors

Table 3 Hyperparameters for training bag-of-words based approaches

Algorithm	Hyperparameter	Value
k-NN	neighbors	5
	distance measure	Cosine Similarity
NB	smoothing parameter	1
SVM	svm type	nu-SVC
	kernel type	radial basis function (rbf)
	gamma	0
	nu	0.5
	cache size	80
	epsilon	0.001
	shrinking	True
	weight	1

based on TF-IDF are generated by mapping unigrams to 880 features used in training k-NN, NB, and SVM. The sequence lengths used for the evaluation of BERT classifiers are the same as the sequence lengths used to fine-tune the respective BERT base model. For example, a BERT base model fine-tuned with a sequence length of 64 is also evaluated with a sequence length of 64. The evaluation metrics for classification models used in the literature differ significantly. As a result, appropriate metrics must be used based on the problem domain and the characteristics of the dataset, such as balanced or imbalanced. The evaluation metrics used in this study to evaluate classifier performance are accuracy, precision, recall, and F1 score. Finally, the effectiveness of the fine-tuned BERT base model in the classification of helpful and unhelpful reviews is evaluated by comparing it to k-NN, NB, and SVM.

4 Results and Discussion

This section presents and discusses the evaluation results of six fine-tuned BERT classification models using varying sequence lengths. Furthermore, the evaluation results for bag-of-words based classifiers such as k-NN, NB, and SVM are presented and discussed to analyze and report the effectiveness of the different approaches used in this study. Table 4 summarizes the overall results of training and validation of fine-tuned BERT classification models for six different sequence lengths used in this study. For each sequence length, the results include batch size, training loss, validation loss, validation accuracy, training time, and validation time for four epochs.

The training and validation loss for fine-tuned BERT with sequence lengths of 64, 128, 256, 320, 384, and 512 are shown in Fig. 2. The sequence length of 64 is initially used to fine-tune BERT base model for training and validating the classifier. The training and validation loss for a classifier with a sequence length of 64 are mapped in Fig. 2(a). In contrast to the best validation accuracy obtained in epoch 3, the continuously increasing validation loss suggests that additional training will lead to overfitting. The training and validation results of a classifier with a sequence length of 128 show that the training loss decreases from 0.62 in epoch 1 to 0.4 in epoch 4. In contrast, the validation loss decreased from 0.56 in epoch 1 to 0.54 in epoch 2. Following that, the validation loss increased to 0.58 in epoch 3 and 0.61 in epoch 4. Fig. 2(b) shows the training and validation loss for a classifier with a sequence length of 128.

In Fig. 2c, the training and validation loss of a classifier trained with a sequence length of 256 are plotted across four epochs. The results show that training loss decreases from 0.6 in epoch 1 to 0.37 in epoch 4. Furthermore, the validation loss decreases from 0.57 in epoch 1 to 0.56 in epoch 2. The validation loss increases with each epoch, reaching 0.6 in epoch 3 and 0.63 in epoch 4. The training and validation loss of a classifier with a sequence length of 320 over four epochs is shown in Fig. 2d. The training loss is seen to decrease from 0.61 in epoch 1 to 0.4 in epoch 4. However, the validation loss increases from 0.6 in epoch 1 to 0.63 in epoch 2, 0.61 in epoch 3, and finally 0.67 in epoch 4.

The training and validation loss of a classifier with a sequence length of 384 is depicted in Fig. 2e. The training loss is 0.6, 0.56, 0.47, and 0.38 for epoch 1,

Table 4 Training and validation results for the BERT classifier

Sequence Length	Batch Size	Epoch	Training Loss	Valid. Loss	Valid. Accur.	Training Time	Validation Time
64	32	1	0.65	0.63	0.64	0:01:28	0:00:03
		2	0.56	0.65	0.63	0:01:27	0:00:03
		3	0.45	0.72	0.65	0:01:27	0:00:03
		4	0.35	0.75	0.65	0:01:27	0:00:03
128	32	1	0.62	0.56	0.73	0:05:00	0:00:12
		2	0.56	0.54	0.73	0:04:59	0:00:12
		3	0.48	0.58	0.73	0:04:59	0:00:12
		4	0.4	0.61	0.73	0:04:59	0:00:12
256	32	1	0.6	0.57	0.71	0:09:19	0:00:23
		2	0.55	0.56	0.72	0:09:19	0:00:23
		3	0.47	0.6	0.71	0:09:19	0:00:23
		4	0.37	0.63	0.72	0:09:19	0:00:23
320	32	1	0.61	0.6	0.69	0:07:37	0:00:19
		2	0.56	0.63	0.67	0:07:40	0:00:19
		3	0.49	0.61	0.7	0:07:38	0:00:19
		4	0.4	0.67	0.7	0:07:39	0:00:19
384	16	1	0.6	0.57	0.73	0:08:31	0:00:21
		2	0.56	0.54	0.72	0:08:30	0:00:21
		3	0.47	0.57	0.75	0:08:30	0:00:21
		4	0.38	0.64	0.72	0:08:30	0:00:21
512	8	1	0.6	0.55	0.73	0:12:05	0:00:30
		2	0.54	0.6	0.72	0:12:13	0:00:30
		3	0.4	0.86	0.68	0:12:13	0:00:30
		4	0.26	1.07	0.7	0:12:12	0:00:30

epoch 2, epoch 3, and epoch 4, respectively. In epoch 1, the validation loss is 0.57, but it is reduced to 0.54 in epoch 2. The validation loss then increases to 0.57 in epoch 3 and 0.64 in epoch 4. Finally, Fig. 2f depicts the training and validation loss of a classifier with a sequence length of 512. The training loss gradually decreases from 0.6 in epoch 1 to 0.26 in epoch 4. In contrast, the validation loss increased steadily from 0.55 in epoch 1 to 1.07 in epoch 4.

When the training and validation results of classifiers with six different sequence lengths are compared, it is discovered that the training loss decreases in a similar pattern over four epochs. The validation loss, on the other hand, is completely random and has no pattern. The validation accuracy varies at random as well. The best validation accuracy of 0.75 is obtained with a sequence length of 384. In contrast, a sequence length of 64 results in the lowest validation accuracy of 0.64. The results show that as the sequence length increases, so does the time required for training and validation. However, the variation in training

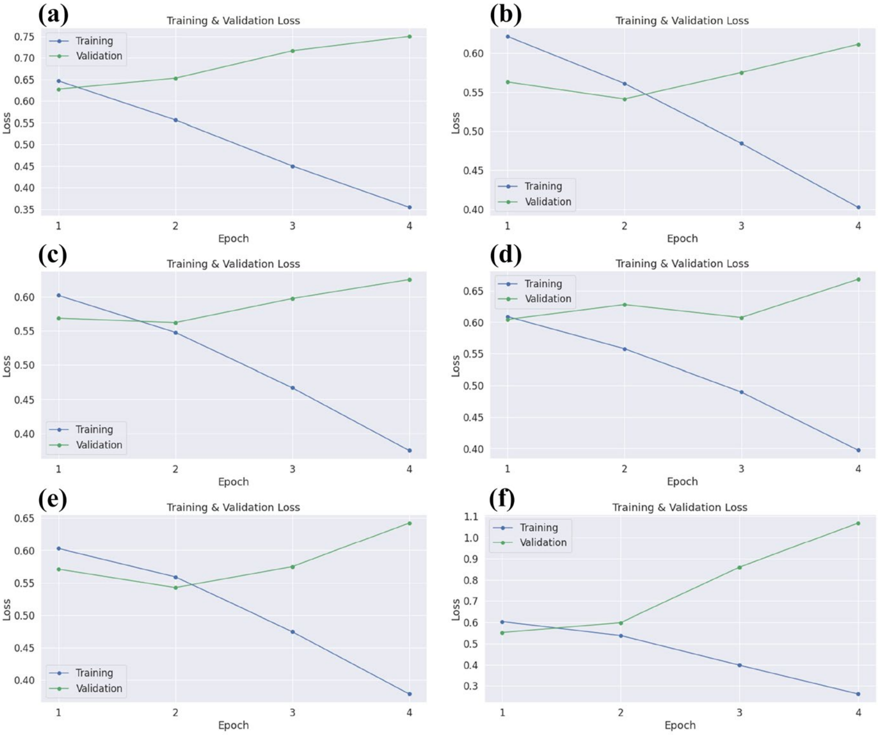


Fig. 2 Training and Validation loss for sequence length **a** 64 **b** 128 **c** 256 **d** 320 **e** 384 **f** 512

and validation times for a classifier with a sequence length of 256 is caused by changes in the performance of the GPU assigned automatically by Google Colab.

The training dataset of 8000 reviews is also used to train bag-of-words based k-NN, NB, and SVM classifiers. Following that, the performance of k-NN, NB, SVM, and six BERT classifiers is evaluated using a test dataset containing 2000 reviews. The prediction results for all classifiers used in this study are summarized in Table 5, where TN, FN, TP, and FP denote true negative, false negative, true positive, and false positive, respectively. According to the evaluation results, the k-NN predicts 584 TP, 416 FN, 608 TN, and 392 FP. The NB predicts 645 TP, 355 FN, 589 TN, and 411 FP. In contrast, SVM predicts 679 TP, 321 FN, 678 TN, and 322 FP. The results show that the BERT classifier with a sequence length of 64 correctly predicts 725 TP and 610 TN samples. In contrast, 390 samples are predicted to be FP, while 275 are predicted to be FN.

According to the comparison of predicted labels with actual labels for the BERT classifier with a sequence length of 128, 703 samples are TP, 684 samples are TN, 316 samples are FP, and 297 samples are FN. The predictions of BERT classifier with a sequence length of 256 yield 735 TP samples, 649 TN samples, 351 FP samples, and 265 FN samples. It is also seen that the BERT classifier with a sequence length of 320 predicts 743 samples as TP, 671 samples as TN, 329 samples as FP and 257 samples as FN. With a sequence length of 384, the BERT

Table 5 Summary of prediction results using the test dataset

Classifier	TP	FN	TN	FP
k-NN	584	416	608	392
NB	645	355	589	411
SVM	679	321	678	322
BERT-64	725	275	610	390
BERT-128	703	297	684	316
BERT-256	735	265	649	351
BERT-320	743	257	671	329
BERT-384	711	289	654	346
BERT-512	666	334	727	273

classifier predicts 711 samples to be TP and 654 samples to be TN. In comparison, 346 samples are predicted to be FP and 289 samples are predicted to be FN. Finally, the results for the BERT classifier with a maximum sequence length of 512 show that 666 of the predicted samples are TP, 727 are TN, 273 are FP, and 334 are FN.

Table 6 shows the evaluation matrices for all classifiers, which include accuracy, precision, recall, and F1 score, calculated using the prediction results given in Table 5. The evaluation results for the bag-of-words based classifiers show that k-NN has the lowest accuracy (0.596) and the lowest F1 score (0.591). In comparison to k-NN, NB performs slightly better, with an accuracy of 0.617 and an F1 score of 0.628. It is interesting to note that SVM outperformed k-NN and NB in terms of accuracy (0.679) and F1 score (0.678). According to the BERT classifier results, the classifier with the shortest sequence length of 64 has the lowest accuracy (0.668) and the F1 score (0.685). In contrast, the classier with a sequence length of 320 achieves the highest accuracy (0.707) and F1 score (0.717). It can also be seen that sequence lengths of 128 and 256 produce competitively better results than sequence lengths of 384 and 512. The results show that the sequence length used to fine-tune and evaluate the BERT base model has a significant impact on classification performance.

Table 6 Evaluation performance of classifiers on the test dataset

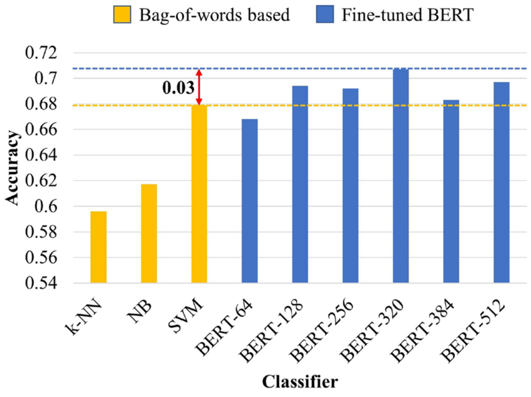
Classifier	Accuracy	Precision	Recall	F1 Score
k-NN	0.596	0.598	0.584	0.591
NB	0.617	0.611	0.645	0.628
SVM	0.679	0.678	0.679	0.678
BERT-64	0.668	0.65	0.725	0.685
BERT-128	0.694	0.69	0.703	0.696
BERT-256	0.692	0.677	0.735	0.705
BERT-320	0.707	0.693	0.743	0.717
BERT-384	0.683	0.673	0.711	0.691
BERT-512	0.697	0.709	0.666	0.687

The accuracy of bag-of-words based classifiers and fine-tuned BERT classifiers is compared in Figure 3. The comparison shows that fine-tuned BERT classifiers outperform bag-of-words-based classifiers. SVM has the highest accuracy and F1 score when compared to other bag-of-words based classifiers (k-NN and NB). SVM achieves slightly higher accuracy than BERT-64 (BERT classifier fine-tuned and evaluated using a sequence length of 64), while the F1 score of SVM is slightly lower than BERT-64. The BERT-320 classifier achieves the highest accuracy of 0.707, which is 0.03 (3%) higher than the accuracy of the SVM classifier, which is 0.679.

When the prediction results of SVM and BERT-320 in Table 5 are closely examined, it is clear that the difference in the performance of both classifiers is due to TP and FN predictions. SVM predicted 679 TP and 321 FN, whereas BERT-320 predicted 743 TP and 257 FN. Only a small difference exists between SVM and BERT-320 TN and FN predictions. The good performance of BERT-320 can be attributed to BERT features as well as the length of the sequence of reviews used to fine-tune and evaluate the classifier. In contrast to bag-of-words-based features, which remove the majority of the words and do not take into account contextual placement of words, BERT features record bi-directional context of words and do not remove any stop words.

Although SVM outperforms k-NN and NB and has comparable accuracy to BERT classifiers, it has been reported that SVM does not perform well on large datasets [59]. Bag-of-words based classifiers require mandatory textual data pre-processing before creating word vectors with TF-IDF to achieve better results. The other problem with bag-of-words based approaches is that it produces a huge list of features from which important features needed to be filtered using some automatic feature selection techniques. Like in this study, 18982 bag-of-words based features are generated from which only 880 features are selected. In contrast, BERT did not require any pre-processing of textual data and converted textual data into specific input format using an uncased BERT tokenizer. Moreover, BERT uses bidirectional representations that condition jointly on the left and right context of a word which is not considered in bag-of-words based approaches.

Fig. 3 Performance comparison of bag-of-words based and fine-tuned BERT classifiers



5 Implications

The findings of this study have several theoretical and practical implications. This study helps researchers in overcoming the contradictory findings of previous studies on the effectiveness of the BERT model to predict review helpfulness by comparing the performance of BERT and bag-of-words-based approaches. The experimental results and performance comparison of BERT with bag-of-words-based approaches will also help practitioners in selecting the best approach. The structure of the dataset used in this study reveals that the average length of helpful reviews is significantly greater than that of unhelpful reviews. This will guide reviewers to use approximately 190 words to write helpful reviews. Prior research had only concentrated on fine-tuning BERT with a fixed sequence length. Hence, six different sequence lengths are used in this study to fine-tune the BERT model and provide insights that will help researchers understand and analyze the significance and impact of using different sequence lengths on classifier predictive performance. Furthermore, the BERT classifier performed best when fine-tuned with a sequence length of 320, indicating to researchers and practitioners that trimming the length of reviews to 320 and using it for prediction produces good results. It may, however, vary depending on the structure of the dataset. The generalized approach presented in this study, which does not rely on any pre-processing or handcrafted features, will assist researchers in better understanding and improving the research being conducted to predict the helpfulness of online customer reviews. Following COVID-19, a new revolution in online business and shopping has emerged. This study provides insights for review platforms to better organize online customer reviews, assisting potential buyers in accessing the quality of products or businesses and making purchase decisions.

6 Conclusion

The volume of online customer reviews is constantly increasing, outpacing humans' cognitive abilities to sort helpful reviews for purchase decisions. The goal of this study is to overcome the limitations of previous studies of handcrafted features, which limit the generalization of the solution. Therefore, BERT, a state-of-the-art technique for various NLU tasks, is used in this study along with three bag-of-words based classifiers (k-NN, NB, and SVM) to predict the helpfulness of online reviews without relying on any handcrafted features. The performance of six BERT classifiers trained with different sequence lengths is evaluated and compared to the performance of bag-of-words based classifiers using a dataset of Yelp shopping reviews. The evaluation results showed that fine-tuned BERT classifiers outperformed bag-of-words-based approaches in classifying helpful and unhelpful shopping reviews. Moreover, the BERT classifier with a sequence length of 320 achieves the highest accuracy and F1 score.

There are a few limitations attached to this study. First, only Yelp reviews are used. Depending on the review platform, the length of reviews may vary. As a result, future work will consider evaluating BERT-based classifiers with varying sequence lengths on datasets of reviews from other review platforms such as Amazon, TripAdvisor, etc. Second, this study only fine-tunes the BERT base model for predicting review helpfulness. In future work, other variants of BERT such as BERT large model, A Robustly Optimized BERT Pretraining Approach (RoBERTa) [60], A Lite BERT (ALBERT) [61] and A Multi-grained BERT (AMBERT) [62] will be fine-tuned and compared to the BERT base model for classification of helpful and unhelpful reviews. This research has theoretical as well as practical implications. This study has theoretical as well as practical implications. This research contributes to the body of knowledge by examining the effect of fine-tuning the BERT base model with different sequence lengths on predicting the helpfulness of reviews. This will help researchers and practitioners understand how shortening reviews to a specific length can improve predictive performance. Furthermore, the structure of the dataset reveals the average length of helpful reviews, which will help reviewers in writing more helpful reviews. Additionally, review platforms can use this BERT-based review helpfulness prediction approach to help customers in overcoming information overload by automatically classifying helpful and unhelpful reviews with acceptable accuracy without relying on any handcrafted features.

Acknowledgements This research was supported by the FAST National University of Computer and Emerging Sciences with a Faculty Research Support Grant (Fall-2021) under the letter No. “11-71-5/NU-R/21”.

Data Availibility Statement The training and test datasets used in this study are available at this link.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Meneghello, J., Thompson, N., Lee, K., Wong, K. W., & Abu-Salih, B. (2020). Unlocking social media and user generated content as a data source for knowledge management. *International Journal of Knowledge Management (IJKM), 16*(1), 101–122.
2. Watanabe, T., Omori, Y., et al. (2020). Online consumption during the covid-19 crisis: Evidence from Japan. *Covid Economics, 38*(16), 218–252.
3. Guo, J., Wang, X., & Wu, Y. (2020). Positive emotion bias: Role of emotional content from online customer reviews in purchase decisions. *Journal of Retailing and Consumer Services, 52,* 101891.
4. Chen, A., Lu, Y., & Wang, B. (2017). Customers’ purchase decision-making process in social commerce: A social learning perspective. *International Journal of Information Management, 37*(6), 627–638.
5. Tata, S. V., Prashar, S., & Gupta, S. (2020). An examination of the role of review valence and review source in varying consumption contexts on purchase decision. *Journal of Retailing and Consumer Services, 52,* 101734.
6. Zhao, Y., Xu, X., & Wang, M. (2019). Predicting overall customer satisfaction: Big data evidence from hotel online textual reviews. *International Journal of Hospitality Management, 76,* 111–121.

7. Yelp. (2021). Fast facts. https://www.yelp-press.com/company/fast-facts/default.aspx, Retrieved June 04, 2021, from https://www.yelp-press.com/company/fast-facts/default.aspx
8. Lee, I. (2017). Big data: Dimensions, evolution, impacts, and challenges. *Business Horizons, 60*(3), 293–303.
9. Bilal, M., Gani, A., Lali, M. I. U., Marjani, M., & Malik, N. (2019). Social Profiling: A review, taxonomy, and challenges. *Cyberpsychology, Behavior, and Social Networking, 22*(7), 433–450.
10. Hf, H., & Krishen, A. S. (2019). When is enough, enough? Investigating product reviews and information overload from a consumer empowerment perspective. *Journal of Business Research, 100,* 27–37.
11. Roetzel, P. G. (2019). Information overload in the information age: A review of the literature from business administration, business psychology, and related disciplines with a bibliometric approach and framework development. *Business Research, 12*(2), 479–522.
12. Lee, S., & Choeh, J. Y. (2018). The interactive impact of online word-of-mouth and review helpfulness on box office revenue. Management Decision
13. Li, M., Huang, L., Tan, C. H., & Wei, K. K. (2013). Helpfulness of online product reviews as seen by consumers: Source and content features. *International Journal of Electronic Commerce, 17*(4), 101–136.
14. Zhu, Y., Liu, M., Zeng, X., & Huang, P. (2020). The effects of prior reviews on perceived review helpfulness: A configuration perspective. *Journal of Business Research, 110,* 484–494.
15. Kim, S. M., Pantel, P., Chklovski, T., & Pennacchiotti, M. (2006). Automatically assessing review helpfulness. In *Proceedings of the 2006 conference on empirical methods in natural language processing, association for computational linguistics* (pp. 423–430).
16. Yang, Y., Yan, Y., Qiu, M., & Bao, F. (2015). Semantic analysis and helpfulness prediction of text for online product reviews. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing* (Volume 2: Short Papers, pp. 38–44).
17. Tang, J., Gao, H., Hu, X., & Liu, H. (2013). Context-aware review helpfulness rating prediction. In *Proceedings of the 7th ACM conference on recommender systems*, ACM (pp. 1–8).
18. Bilal, M., Marjani, M., Hashem, I. A. T., Gani, A., Liaqat, M., & Ko, K. (2019). Profiling and predicting the cumulative helpfulness (Quality) of crowd-sourced reviews. *Information, 10*(10), 295.
19. Bilal, M., Marjani, M., Hashem, I. A. T., Abdullahi, A. M., Tayyab, M., & Gani, A. (2019). Predicting helpfulness of crowd-sourced reviews: A survey. *2019 13th International conference on mathematics*. Computer Science and Statistics (MACS), IEEE: Actuarial Science (pp. 1–8).
20. Bilal, M., Marjani, M., Lali, M. I., Malik, N., Gani, A., & Hashem, I. A. T. (2020). Profiling users' behavior, and identifying important features of review "helpfulness''. *IEEE Access, 8,* 77227–77244.
21. Diaz, G. O., & Ng, V. (2018). Modeling and prediction of online product review helpfulness: A survey. In *Proceedings of the 56th annual meeting of the association for computational linguistics* (Volume 1: Long Papers, pp. 698–708).
22. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. Preprint arXiv:181004805.
23. Wu, S. H., & Chen, Y. K. (2020). Cross-domain helpfulness prediction of online consumer reviews by deep learning model. In *2020 IEEE 21st international conference on information reuse and integration for data science (IRI)*, IEEE (pp. 412–418)
24. Xu, S., Barbosa, S. E., & Hong, D. (2020). Bert feature based model for predicting the helpfulness scores of online customers reviews. In *Future of information and communication conference*, Springer (pp. 270–281).
25. Liu, Y., Huang, X., An, A., & Yu, X. (2008). Modeling and predicting the helpfulness of online reviews. In *2008 Eighth IEEE international conference on data mining*, IEEE (pp. 443–452).
26. Lee, S., & Choeh, J. Y. (2014). Predicting the helpfulness of online reviews using multilayer perceptron neural networks. *Expert Systems with Applications, 41*(6), 3041–3046.
27. Krishnamoorthy, S. (2015). Linguistic features for review helpfulness prediction. *Expert Systems with Applications, 42*(7), 3751–3759.
28. Hu, Y. H., & Chen, K. (2016). Predicting hotel review helpfulness: The impact of review visibility, and interaction between hotel stars and review ratings. *International Journal of Information Management, 36*(6), 929–944.
29. Singh, J. P., Irani, S., Rana, N. P., Dwivedi, Y. K., Saumya, S., & Roy, P. K. (2017). Predicting the "helpfulness'' of online consumer reviews. *Journal of Business Research, 70,* 346–355.

30. Hu, Y. H., Chen, K., & Lee, P. J. (2017). The effect of user-controllable filters on the prediction of online hotel reviews. *Information & Management, 54*(6), 728–744.
31. Chen, C., Yang, Y., Zhou, J., Li, X., & Bao, F. (2018). Cross-domain review helpfulness prediction based on convolutional neural networks with auxiliary domain discriminators. In *Proceedings of the 2018 conference of the north american chapter of the association for computational linguistics: Human language technologies*, (Volume 2, Short Papers, pp. 602–607).
32. Zhang, Y., & Lin, Z. (2018). Predicting the helpfulness of online product reviews: A multilingual approach. *Electronic Commerce Research and Applications, 27,* 1–10.
33. Akbarabadi, M., & Hosseini, M. (2018). Predicting the helpfulness of online customer reviews: The role of title features. *International Journal of Market Research, 62,* 1470785318819979.
34. Ma, Y., Xiang, Z., Du, Q., & Fan, W. (2018). Effects of user-provided photos on hotel review helpfulness: An analytical approach with deep leaning. *International Journal of Hospitality Management, 71,* 120–131.
35. Saumya, S., Singh, J. P., Baabdullah, A. M., Rana, N. P., & Dwivedi, Y. K. (2018). Ranking online consumer reviews. *Electronic Commerce Research and Applications, 29,* 78–89.
36. Lee, P. J., Hu, Y. H., & Lu, K. T. (2018). Assessing the helpfulness of online hotel reviews: A classification-based approach. *Telematics and Informatics, 35*(2), 436–445.
37. Sun, X., Han, M., & Feng, J. (2019). Helpfulness of online reviews: Examining review informativeness and classification thresholds by search products and experience products. *Decision Support Systems, 124,* 113099.
38. Olatunji, I. E., Li, X., & Lam, W. (2019). Context-aware helpfulness prediction for online product reviews. In *Asia information retrieval symposium*, Springer (pp. 56–65).
39. Du, J., Rong, J., Michalska, S., Wang, H., & Zhang, Y. (2019). Feature selection for helpfulness prediction of online product reviews: An empirical study. *PloS one, 14*(12), e0226902.
40. Ge, S., Qi, T., Wu, C., Wu, F., Xie, X., & Huang, Y. (2019). Helpfulness-aware review based neural recommendation. *CCF Transactions on Pervasive Computing and Interaction, 1*(4), 285–295.
41. Chen, C., Qiu, M., Yang, Y., Zhou, J., Huang, J., Li, X., & Bao, F. S. (2019). Multi-domain gated cnn for review helpfulness prediction. In *The world wide web conference* (pp. 2630–2636).
42. Luo, Y., & Xu, X. (2019). Predicting the helpfulness of online restaurant reviews using different machine learning algorithms: A case study of yelp. *Sustainability, 11*(19), 5254.
43. Saumya, S., Singh, J. P., & Dwivedi, Y. K. (2019). Predicting the helpfulness score of online reviews using convolutional neural network. *Soft Computing, 24,* 1–17.
44. Fan, M., Feng, C., Guo, L., Sun, M., & Li, P. (2019). Product-aware helpfulness prediction of online reviews. In *The world wide web conference* (pp. 2715–2721).
45. Kong, L., Li, C., Ge, J., Ng, V., & Luo, B. (2020). Predicting product review helpfulness a hybrid method. *IEEE Transactions on Services Computing*.
46. Son, J., Negahban, A., Lee, Y., Connolly, J., & Chiang, D. (2020). When more is more and less is more: Depth and breadth of product reviews and their effects on review helpfulness. In *Proceedings of the 53rd Hawaii international conference on system sciences*.
47. Malik, M., & Hussain, A. (2020). Exploring the influential reviewer, review and product determinants for review helpfulness. *Artificial Intelligence Review, 53*(1), 407–427.
48. Du, J., Zheng, L., He, J., Rong, J., Wang, H., & Zhang, Y. (2020). An interactive network for end-to-end review helpfulness modeling. *Data Science and Engineering, 5*(3), 261–279.
49. Namvar, M. (2020). A novel approach to predict the helpfulness of online reviews. In *Proceedings of the 53rd Hawaii international conference on system sciences*.
50. Malik, M. S. I. (2020). Predicting users' review helpfulness: The role of significant review and reviewer characteristics. *Soft Computing, 24,* 1–16.
51. Bilal, M., Marjani, M., Hashem, I. A. T., Malik, N., Lali, M. I. U., & Gani, A. (2021). Profiling reviewers' social network strength and predicting the "helpfulness'' of online customer reviews. *Electronic Commerce Research and Applications, 45,* 101026.
52. Mauro, N., Ardissono, L., & Petrone, G. (2021). User and item-aware estimation of review helpfulness. *Information Processing and Management, 58*(1), 102434.
53. Du, J., Rong, J., Wang, H., & Zhang, Y. (2021). Neighbor-aware review helpfulness prediction. *Decision Support Systems, 148,* 113581.
54. Lee, M., Kwon, W., & Back, K. J. (2021). Artificial intelligence for hospitality big data analytics: Developing a prediction model of restaurant review helpfulness for customer decision-making. *International Journal of Contemporary Hospitality Management.*

55. Lutz, B., Pröllochs, N., & Neumann, D. (2022). Are longer reviews always more helpful? Disentangling the interplay between review length and line of argumentation. *Journal of Business Research, 144,* 888–901.
56. Kashyap, R., Kesharwani, A., & Ponnam, A. (2022) Measurement of online review helpfulness: A formative measure development and validation. *Electronic Commerce Research* 1–34.
57. Olmedilla, M., Martínez-Torres, M. R., & Toral, S. (2022). Prediction and modelling online reviews helpfulness using 1d convolutional neural networks. *Expert Systems with Applications, 198,* 116787.
58. Yelp. (2019). Yelp open dataset. https://www.yelp.com/dataset, Retrieved March 30, 2021, from https://www.yelp.com/dataset.
59. Jx, D., Krzyzak, A., & Suen, C. Y. (2005). Fast svm training algorithm with decomposition on very large data sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(4), 603–618.
60. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. Preprint arXiv:19071 1692.
61. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. Preprint arXiv:190911942.
62. Zhang, X., Li, P., & Li, H. (2020). Ambert: A pre-trained language model with multi-grained tokenization. Preprint arXiv:200811869.

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