National College of Ireland

Leveraging Transfer learning techniques- BERT, RoBERTa, ALBERT and DistilBERT for Fake Review Detection

MSc Research Project
Data Analytics

Priyanka Gupta
Student ID: x19223030

School of Computing
National College of Ireland

Supervisor: Bharathi Chakravarthi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Priyanka Gupta
Student ID:	x19223030
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Bharathi Chakravarthi
Submission Due Date:	16-08-2021
Project Title:	Leveraging Transfer learning techniques- BERT, RoBERTa, ALBERT and DistilBERT for Fake Review Detection
Word Count:	6933
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Leveraging Transfer learning techniques- BERT, RoBERTa, ALBERT and DistilBERT for Fake Review Detection

Priyanka Gupta
x19223030

Abstract

In this era of the internet, the online review system has grown tremendously, where customers share their first-hand experiences about the products or services. These reviews influence the purchasing decision of future customers and have a positive or negative financial impact on businesses. Spam reviews are written with an agenda to promote or demote a business and mislead the customers. Hence to maintain the integrity of the online review system, it is crucial to detect fake reviews. To overcome the limitations of traditional machine learning and neural network-based models, we have leveraged transfer learning and used transformer-based pre-trained models BERT, RoBERTa, ALBERT, and DistilBERT to build fake review classifier. Performance of all the models is evaluated, considering accuracy and weighted F1-source as the primary metric for evaluation. The classifier produced using RoBERTa has outperformed the baseline model in detecting fake reviews.

1 Introduction

1.1 Background & Motivation

In this era of the Internet, online platforms are widely used for numerous activities such as hotel reservations, product purchasing, movie ticket booking, and many more. With the rise in online platforms, the number of people posting their opinions on these platforms has also seen a sharp rise. These reviews describe the first-hand experience of the reviewer and become a great source of information for any future customers. A large number of customers rely on online reviews to understand about services or products making any decisions. These reviews not only influence potential buyers but also provide insights to businesses that assist in increasing customer satisfaction. However, customers and businesses can benefit from these reviews only if they are credible. As reviews have a positive or negative financial impact on the business, few companies hire spammers to promote their brand or criticize their rivals' brands for influencing sales patterns. More than ever, deceptive reviews are posted on platforms such as Yelp and Amazon, which damages the online review system. A recent study has shown that 80% of customers tend not to buy products with high negative reviews (Tang and Cao; 2020). Hence, detecting fake reviews which mislead or deceive the customers becomes a crucial area for research.

Research around spam review detection was first explored by Jindal and Liu (2008) in the year 2008. The reviews of the Amazon dataset were subdivided into three categories. The first category consists of untruthful opinions written by spammers without using the product or service and has either positive or negative notions. The brand-only reviews that either promote or defame the brands and have no relation with the product were considered the second category. The last category includes advertisements or irrelevant opinions. A classifier built using logistic regression algorithm, and thirty-six features about the text, users, and product were used to detect span reviews.

With advancements in technology, researchers have used Natural language processing (NLP) for text classification problems, including fraud reviews detection, to build a reliable online review system. As per literature, remarkable results are achieved to detect fraudulent reviews using Support Vector Network, Naive Bayes (Li et al.; 2021), LSTM (Wang et al.; 2018), CNN Bi-LSTM (Liu et al.; 2020), and other machine learning and deep learning-based algorithms. However, the machine learning-based models depend on domain-specific knowledge for manual feature extraction, and deep learning models require massive labeled datasets for training. Also, these models are only trained on domain-specific data, which causes their performance to lack generalization. Literature review reveals that transformer-based pre-trained models such as BERT, RoBERTa, ALBERT, and DistilBERT, achieved state-of-the-art performance to produce text classification models. However, they are not yet thoroughly explored for spam review detection.

1.2 Research Question & Objective

1.2.1 Research Question

To what extent the pre-trained models of BERT family be leveraged for developing a generalized fake review detection system?

1.2.2 Research Objective & Contribution

The major contribution of this research is the identification of fake reviews present on various platforms by building an efficient, generalized, and robust model using transfer learning techniques and utilizing minimum computational resources. Below mentioned are the objectives and contributions to address the research question outlined above –

1. A critical analysis and investigation of research performed for fake review identification.

2. Data cleaning and pre-processing to meet the input parameter requirement of the transfer learning models: BERT, RoBERTa, ALBERT, and DistilBERT.

3. Implementation of BERT, RoBERTa, ALBERT, and DistilBERT model for fake review classification task for various dataset samples to achieve reasonable results by using small labeled dataset and limited computational power.

4. Performance evaluation and comparison of all applied models with accuracy and f1-score as the metric.

5. Comparison of trained models with a baseline model.

1.3 Roadmap

The rest of this report is organized as follows: The following sections discuss the existing literature produced for fake review detection, followed by the methodology adopted for this research project. The subsequent section discusses the project's architecture and components involved. In the next section, the project's implementation details are explained, followed by a discussion of the results. In the end, the conclusion and future work is discussed.

2 Related Work

Jindal and Liu were among the first researchers to investigate fake review detection in 2008 (Jindal and Liu; 2008). In the past decade, the spam content on online platforms has increased significantly. This has motivated the researchers to build a steady system that can identify these spam reviews with reasonable accuracy. Over the years, various methods are explored by researchers. In this section, we have discussed the literature published about all such techniques, including Rule-based, Graph-based, Machine learning-based, and Deep learning-based methods.

2.1 Rule-based Methods

Multiple researchers adopt rule-based methods to detect fake reviews. The performance of such models depends upon the quality of manual feature rules defines based on domain expertise and knowledge. The efficiency of the model to detect spam reviews is thwarted if the declared set of rules are inadequate, as these predefined rules assist in finding significant words from reviews that express an opinion and then provide sentiment polarity

by aggregating such words with opinions (Shan et al.; 2021).

In the initial days of research about the fake review detection system, Lim et al. (2010) used a rule-based approach to detect spammers using the Amazon review dataset. They studied rating behaviour patterns to identify abnormal behaviour and detect spammers. Similarly, researchers in Hu, Bose, Gao and Liu (2011) and Hu, Liu and Sambamurthy (2011) used data from Amazon and Barnes Noble and defined rules such as numbers of books evaluated in a given time to detect spammers. Researchers in Sam and Chatwin (2015) built a spam detection model for reviews related to electronic commodities posted on social networking websites. They used a rule-based approach that analysed emotions and keywords of posts to detect fake reviews with over 90% accuracy.

Recently, researchers used a white-box rule-based model consisting of eight rules to build a spam detection system (Jnoub and Klas; 2020). This overcomes the black-box nature of machine learning and deep learning models and provides coherent information about the uses of personal data. Also, making changes to the existing system is more accessible. Model built for movie review dataset achieved competitive results by using parameters such as review quantity and time, number of dislikes, and more.

2.2 Graph-based Methods

The graph-based spam review detection model is another technique used by researchers. It uses iterative calculations to classify fake reviews. Researchers in Wang et al. (2012) built a novel method in which relations between stores, reviews, and all reviewers are used to build a heterogeneous review graph. These intricate relationships captured in the graph are used to detect fraudulent reviews. By plotting a social graph using the peculiar behaviour of reviewers, a robust system can de develop that is difficult for spammers to circumvent (Tang and Cao; 2020). Recently, an unsupervised graph-based model was developed in Bidgolya and Rahmaniana (2020) to detect spam reviews. For multiple deception scenarios, trust, reliability, and honesty values were estimated for reviewers, products, and reviews to plot graph nodes. Evaluation of the built model shows that it outperformed other graph-based models developed for spam detection. However, the efficiency of such models depends on the domain knowledge as its performance decreases if all deceptive scenarios are not considered. To overcome this problem, researchers prefer using machine learning or deep learning-based models for spam detection.

2.3 Machine learning-based Methods

Researchers widely use machine learning techniques to build better-performing spam models. In the past decade, researchers have explored various supervised learning, semi-supervised and unsupervised machine learning techniques. In this section, the research conducted using machine learning techniques are elaborated.

In Jindal and Liu (2008), the authors used three classification algorithms, Random Forest, Naïve Bayes, and support vector machine, to build a fake review detection model. Unigram and Bigram were used for feature extraction and then using stacking, and voting and ensembled model was built for classification. Hernández-Castañeda et al. (2017) used SVM for spam review detection of mixed domain. Word space model, Latent Dirichlet

Allocation, and LIWC were used for feature extraction, and the model was evaluated on DeRev, Opinions, and OpSpam datasets. This domain-independent classifier had an accuracy of 64%. The model did not perform better than the state-of-the-art domain-independent classifiers built using deep neural networks.

Using aspect-oriented sentiment mining, researchers in Li et al. (2021) proposed a novel approach to identify groups of spam reviews backed by nominated topics. First, comparable groups and topics were defined. Using the K-means clustering algorithm, the reviews were divided into clusters. Post which, by considering time burstiness and content duplication, the groups were separated as fake and genuine. The classifier was evaluated by using data from JD.com, and state-of-the-art performance was achieved. However, as the model considers duplicate content to be fake, it may not be reliable for all scenarios. In Wang et al. (2020), the authors used multiple features of the reviews and the reviewers to build a spam detection model. In total, seven machine learning algorithms (DT, NB, LR, SVM, LDA, KDD, and RF) were applied, and their performance was evaluated using the Yelp dataset. Statistical analysis proved that the performance of SVM and RF classifiers were remarkable. More recently, four semi-supervised algorithms, SVM, NB, RF, and Transductive SVM, were explored to build fake review classifier in Ligthart et al. (2021). Features such as Unigrams, Bigrams, POS-tags, TF-IDF values, were used and the model was evaluated on Yelp Chi and AMT datasets. Naive Bayes with self-training algorithms outperformed other classifiers. The researchers in Alsubari et al. (2020) applied three supervised machine learning techniques, Decision tree, Adaptive Boosting, and Random Forest, to detect fake reviews using notion of Information Gain for feature selection. Essential and discriminative features such as authenticity and polarity were selected using LIWC, POS, and sentiment analysis techniques. Adaptive Boosting outperformed other classifiers to detect fake reviews in the electronics domain.

Five machine learning algorithms, NB, LR, KNN, SVM, and RF, were uses to build classifier for fraudulent reviews detection using the Yelp restaurant dataset (Elmogy et al.; 2021). Bi-gram, Tri-gram, and TF-IDF techniques were used to extract reviews' textual features as well as reviewers' behavioural features. The KNN classifier with the value of K as 7 provided a better F1 score than other classifiers. In this work, the authors only considered only three behavioural features. In similar research (Shan et al.; 2021), authors proposed a framework to detect spam reviews using 22 distinct features categories. These extracted features based on content, rating, and language were used to train Classification and Regression Trees (CART), RF, NB, SVM, and MLP classifiers. Proposed models were evaluated using the Yelp dataset. By considering all the inconsistency features, the model showed significant improvement in classifying fake reviews compared to baseline.

2.4 Deep Learning-based Methods

Models based on Neural networks have performed well for natural language processing classification tasks. In recent years, Deep learning-based models such as CNN, RNN, and LSTM are explored by researchers for fake reviews detection. In this section, the research conducted using deep learning techniques are elaborated.

Wang et al. (2018) build a fake review detection model using the long short-term memory

recurrent neural network. The proposed model only used textual features to train the model, and the results were evaluated using the dataset collected from web pages available in Taiwan. The model consists of an input layer, LSTM layer, and output layer. For dimension reduction, hidden layers are present in between. LSTM performed better than the baseline SVM model. In another research (Jain et al.; 2019), the authors proposed two Deep Neural Network (DNN) based approaches, hierarchical CNN-GRN and Multi instant learning (MIL). For n-gram feature extraction, three-layer CNN was employed, and GRN was used to learn semantic dependencies between features. To handle reviews of variable length, the input text is divided into multiple text inputs. The proposed model was evaluated using Yelp Zip, Movie Review, Drug Review, four-city, and Deceptive Spam Corpus datasets, and the model performed better than the baseline RNN and CNN models for all datasets. Hajek et al. (2020) proposed neural network models trained using high-dimensional feature representation to classify reviews into fake and genuine categories. The models were trained using n-grams, lexicon-based emotion indicators, and word embeddings. On similar lines, Guo et al. (2021) proposed a Deep Graph neural network-based Spammer detection (DeG-Spam) model. The model aimed to capture all relations present and separately model occasional relations and inherent relations to compose feature expressions. In comparison with the baseline, the performance of the DeG-Spam model was better.

More recently, Liu et al. (2020) used information provided by fine-grained aspects and implicit patterns inferred between reviews, uses, and products to build fake review classifiers instead of behavioural and textual features. The proposed attention-based Multilevel Interactive Attention Neural Network with Aspect plan (MIANA) model consists of Sentence-level Interactive Attention Neural network module (SIAN) and Word-level Fusion Module (WFM). The model's performance was evaluated using Yelp reviews datasets. In another research work (Bhuvaneshwari et al.; 2021), the attention mechanism is leveraged, and a Self-Attention-based CNN Bi-LSTM (ACB) model is developed to find deceptive reviews. The CNN obtains n-gram features by learning sentence representation. Bi-LSTM combines the sentence vectors as document feature vectors. Finally, by using contextual information, it detects fake reviews achieving state-of-the-art performance.

2.5 Transformers and Language-based Models

Kennedy et al. (2020) used deep learning techniques, LSTM, Feedforward neural network, CNN, along with transfer learning technique, BERT to build fake review classifier. The authors used OpSpam and Yelp datasets for model training and performed an analytic comparison of the models. Also, machine learning models were built using LR and SVM algorithms. To train these machine learning models, features such as Part-of-Speech as a percentage, average length of review, reviewers' features, and more were extracted from the dataset by performing feature engineering. The comparison of all machine learning, deep learning and, transfer learning models revealed BERT to be the best performing model as it achieved competitive state-of-the-art performance for fake review detection when evaluated on OpSpam dataset.

Shan et al. (2021) utilized transformer-based models to detect fake news regarding the COVID-19 pandemic on social media platforms. The researchers observed that the domain-specific language models yield better performance for this sequence classification

task. Experiments using various loss functions are also conducted, and the BCE-Dice Loss function has produced better loss optimization. Models are also built by feeding the pre-trained models embeddings to BiLSTM Layer, which achieved notable performance. CT-BERT Embeddings achieved the highest weighted average F1-score to detect fake news with dense layer architecture.

2.6 Conclusion

Researchers have leveraged various machine learning and deep learning techniques to solve this classification problem and produced remarkable results. The implementation of machine learning models is relatively straightforward and works well with limited computational resources. However, feature engineering is a challenging task for traditional machine learning techniques as it requires features to be extracted from the dataset, which demands domain expertise. Also, such models can not characterize global semantic information of reviews which hampers its spam detection capabilities. On the other hand, Neural networks have proven to produce more robust models for spam detection. Deep learning techniques have omitted the need for manual feature extraction and can extract features directly from the training data, eliminating the need for domain expertise. Deep learning models fail to apprehend long-term dependencies of sequences, even though it uses word embedding to capture semantic meaning. Also, for an efficient fake review detection model, the deep learning models require a large amount of labelled data and significant computational resources. We have used state-of-the-art transfer-learning techniques to build a fake review detection classifier to overcome these issues and achieve significant outcomes. The transformer-based models are pre-trained using humongous data and are not rigorously explored for this area of research.

3 Methodology

This research aims to build a model that is successful in classifying fake or span online reviews. It is vital to use a robust and structured methodology to implement any data mining project successfully. The two generally used approaches for such projects are Knowledge Database Discovery (KDD) and Cross-Industry Standard Process for Data Mining (CRISP-DM). After careful consideration, we have decided to use the CRISP-DM methodology to build the classifier as it best fits the requirement of this project. It permits movement within the phases as and when required, and later stages can benefit from learnings of previous stages. Figure 1 illustrates the CRISP-DM methodology used to build the Fake review classification model, and this section provides a detailed explanation of each stage of the methodology followed.

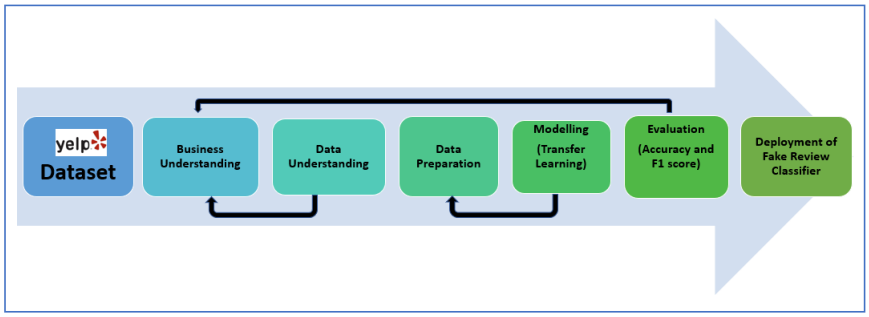

Figure 1: Research methodology for fake review detection

3.1 Business Understanding

In this era of the internet, along with the development of available online services, the trend of posting reviews and opinions on numerous websites has also increased significantly. Websites such as Amazon and Yelp have thousands of customer reviews for each product or service. These customer reviews influence the purchasing decision of the future customer and also help the brand understand the customer's views on the products. In this competitive market, where customers have so many options to choose from, online reviews play a critical role and affect potential buyers' decisions. These reviews have a direct financial impact on the business as positive reviews can increase sales, whereas negative reviews can hamper sales. In the past decade, with the increasing influence of customer reviews on businesses, the number of fake reviews has increased tremendously. Companies hire professionals to write fake reviews to promote their brand and products and write negative reviews to criticize their rival's products. As these deceptive and biased reviews mislead the customers, it is essential to identify such reviews and ensure that such content is not posted on the websites.

3.2 Data Understanding

The Yelp dataset collected by Mukherjee et al. (2013) is used to train the classification model to detect fake reviews. This dataset consists of customer reviews posted on the Yelp website about hotels, restaurants, and other businesses. The dataset consists of two sub-datasets, and together they have over 1.4 million records that were initially annotated or filtered by Yelp using its filtering algorithm. Along with the reviews and their classification as genuine or fake, the dataset also contains ten features about the reviews and thirteen features about the reviewers.

Before implementing any model, performing exploratory data analysis on the dataset is crucial to gain insights and understand patterns present in the dataset. First, the two sub-datasets, 'Yelp hotel reviews' and 'Yelp restaurant reviews,' are merged to create a consolidated dataset. This consolidated dataset is analysed, and the percentage distribution of fake and genuine reviews present in the dataset is plotted in a pie chart represented by Figure 2. The dataset contains 59.7% of genuine reviews and 40.3% of fake reviews. As both the classes have fair representation in the dataset, there is no need for applying any resampling technique. Table 1 illustrates the overview of the dataset.

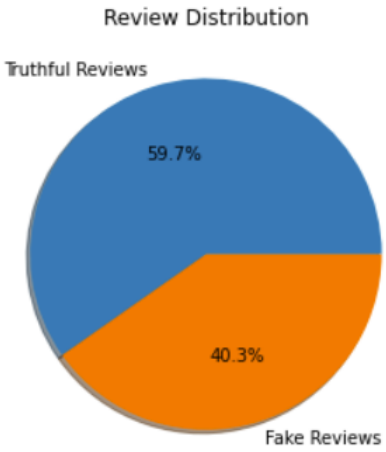

Figure 2: Data distribution

Table 1: Dataset overview

Total number of reviews	**Total reviews**	**Fake reviews**	**Fake reviews (%)**
Restaurant reviews on Yelp	788471	326981	41.47%
Hotel reviews on Yelp	688329	267544	38.86%

3.3 Data Preparation

The quality of data employed to train the model determines the model's performance, making data preparation a significant phase in building the classification model. Reducing the noises present in the data and eliminating the irrelevant data leads towards getting the optimum result. First, the data set is checked for null or blank values, and appropriate steps are taken to hand these values. The columns which will be used to train the model are cleaned by removing punctuations, pieces of HTML tags left in the text, special characters, numbers, and any undesired white spaces. As we are using transformer-based pre-trained models to train the classifier, we are abstaining from using traditional text pre-processing techniques such as stop words removal, stemming, and lemmatization to keep the semantic meanings of the reviews intact. Before feeding the data to pre-trained models, the raw data is converted to a suitable format by using tokenizers specific to the pre-trained model for tokenizing each sentence.

3.4 Modeling

To build a fake review detection model, an in-depth analysis of existing literature is conducted, and various machine learning and deep learning models are taken into consideration. Finally, we have decided to leverage the pre-trained transfer learning models as they have shown exceptional performance for Natural language processing tasks and are not yet been explored to their full potential to detect fake online reviews. Four such models, BERT, RoBERTa, DistilBERT, and ALBERT, are selected for this research work with the objective of building a reasonably accurate model while focusing on minimizing the required computation resources. Fine-tuning of these pre-trained models is conducted on Google Collaborator with GPU enabled using Simple Transformers library. This

library simplifies the implementation of complex transformer-based models and is built on top of the Hugging face library. The implementation of these models is explained in detail in section 5. To evaluate the model's performance, the results are compared with baseline model built by researchers in Mukherjee et al. (2013).

3.5 Evaluation

The data set used to train the models is split into train, validation, and test sets in the ratio of 80:10:10. This 10% data is employed to evaluate the classifier's performance. The accuracy, Precision, Recall, and F1 score values of each model are determined and compared to assess the model's performance.

Accuracy: It measures the correctness of the model. It is defined as the total number of correct predictions made by the model, divided by the total predictions.

Precision: Precision is the percentage of positive predictions from the total predicted positive instances. It calculates the accuracy for the minority class.

Recall: Recall measure the model's capability to detect positive specimens. It provides an indication of missed positive predictions.

F1 score: F1 score is the harmonic mean of precision and recall. The contribution of precision and recall makes it a balanced score, and the model's performance can be considered better with a higher F1 score.

4 Design Specification

Figure 3 displays the architecture of our research. A detailed explanation of each step is provided in Section 5.

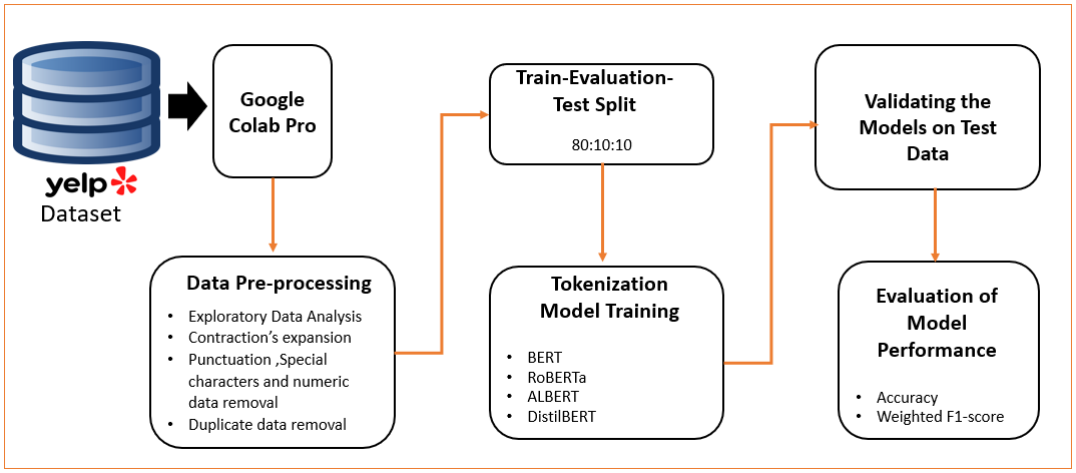

Figure 3: Project Architecture

The remaining section elaborates upon the various pre-trained models its components used in our research.

4.1 Transformer

The Transformers architecture was introduced in the paper' Attention is all you need' by Vaswani et al. (2017). Transformers are built on encoder-decoder architecture. It uses multi-head self-attention mechanism, which enables sequence-to-sequence learning and abolishes the need for recurrent neural networks. The positional encoding of input sequence is done to understand relative position of tokens. In encoder block, multi-head attention layer and feed-forward neural network stacked together, and decoder consists of additional attention layer. During the model training, transformer allows parallelization to enhance performance. The output of decoder is fed into a linear layer followed by a SoftMax layer to predict the target variable.

4.2 Tokenization and Embedding

All the pre-trained models require the input text to be in a particular format. Tokenizers divides the input text, in our case reviews, into smaller units known as tokens. These tokens are then converted into IDs before model training. All the transfer learning models are pre-trained on a specific corpus with fixed vocabulary. The input text given to the model might contain words that are not present in the fixed vocabulary of the model. To handle these out-of-vocabulary (OOV) words, BERT uses a WordPiece tokenizer (Schuster and Nakajima; 2012) that breaks down the words into several subwords to preserve information from the input data. Each pre-trained model has its own tokenizer. Similar to BERT, DistilBERT uses a WordPiece tokenizer. On the other hand, RoBERTa uses a byte-level Byte-Pair-Encoding tokenizer (Sennrich et al.; 2015), and ALBERT uses SentencePiece tokenizer (Kudo and Richardson; 2018).

Before tokenization, in order to help the model distinguish between the sentences,[CLS] and [SEP] tokens are added at beginning and end of sentences. Also, the pre-trained models require all the input text to be of identical length. To accomplish the same, padding is performed based on the max length parameter. If review is smaller than the specified maximum length, then padding(empty tokens) is added. Whereas if the sentence is longer than specified maximum length, then extra tokens are truncated. This tokenized input text of uniform length forms the token embedding layer. Apart from token embedding, there are positional embeddings and a segment embedding layer. Positional embeddings represent the position of words in the input text, and segment embedding differentiates between sentences if input has multiple sentences.

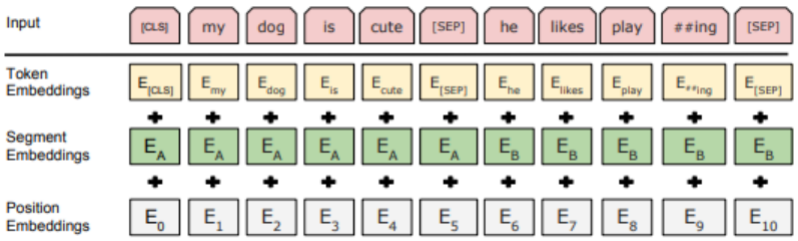

Figure 4: BERT Embedding representation

4.3 BERT

Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based pre-trained model introduced by the Google AI team (Devlin et al.; 2018). It is pre-trained on 2500 million words text paragraphs and 800 million words books corpus of English language, available on Wikipedia. As opposed to sequential learning of textual data performed by directional models from either right or left, BERT utilizes bidirectional learning and learns the context of text by learning right and left word context. BERT has demonstrated to achieve state-of-the-art results for several Natural language processing tasks. Masked Language Modelling (MLM) and Next Sentence Prediction (NSP) are the two tasks performed in the BERT's pretraining phase. BERT architecture was initially introduced in two variants, BERT Base and BERT Large.

For this research work, we are using BERT base architecture which is built on 12 transformer-encoder blocks as the memory and time requirement of the base model is less than BERT large. It is trained on lower-cased English text with a vocabulary size of about 30,000 tokens. Each transformer block has 768 hidden layers and 12 self-attention heads and is trained on 110M parameters. The implementation of the model is explained in-depth in Section 6.

4.4 RoBERTa

The robustly optimized BERT approach (RoBERTa) is a transformer-based pre-trained model developed by the Facebook AI team (Liu et al.; 2019), which builds on BERT's language masking strategy. Using better training methodology, RoBERTa has outperformed BERT in several NLP tasks. It is trained using more data with a larger mini-batch size and longer sequences, for more extended periods. It removed next sentence prediction and hence is able to eliminate the problem of NSP loss present in BERT. RoBERTa is pre-trained on 160 GB corpus of CCNews datasets and English Wikipedia. It employs dynamic masking during training to mask different sets of tokens during each epoch of training. For tokenization, it uses Byte-Pair Encoding (BPE) with a vocabulary size of about 50,000 tokens instead of BERT's character-level BPE. The roberta-base model used in this research work to build the fake review classifier has 12 layers of transformers with 12 attention heads and 768 hidden layers, along with 125 million parameters.

4.5 DistilBERT

A distilled version of the BERT pre-trained model known as DistilBERT was developed by Sanh et al. (2019), which is lighter, cheaper, much smaller, and faster than BERT. Compared to the BERT base model, DistilBERT uses 40% fewer parameters, operates 60% faster, and preserves 97% of the model's performance. Along with low computational and resource requirements, this model also mitigates few other limitations of BERT, such as word piece embedding and fixed input length size problems. DistilBERT has retained 50% of BERT's layers and removed pooler and token-type embeddings, which are present in the architecture of BERT. This model employs a cosine embedding loss, masked language modeling loss, and student loss with the random initialization, i.e., triple loss system. The input text needs to be tokenized, converted into token IDs, and padded before model training. The distilbert-base-uncased model, which was distilled from the bert-base-uncased model, has 6 transformers with 12 self-attention layers, 768 hidden

layers, and 66M parameters. We have selected DistilBERT as one of the models to build the classifier as its limited-resource requirement complies with this project's objective.

4.6 ALBERT

A Lite BERT (ALBERT) is another pre-trained model built on BERT's architecture but altered to overcome its shortcomings, such as longer training time and memory limitations. Introduced by Lan et al. (2019), ALBERT uses two-parameter reduction techniques that lessen GPU/TPU memory usage and boosts the speed of the training process to achieve comparable or more favourable results than BERT. This self-supervised learning model uses Sentence order prediction (SOP) instead of BERT's Next Sentence Prediction, thus eliminating NSP losses. It is trained on English Wikipedia and Books corpus. ALBERT has successfully increased hidden size while keeping the vocabulary embedding's parameter size within a limit, thus converting the large vocabulary embedding matrix into smaller matrices for embedding factorization. The albert-base-v1 pertained model uses 12 repeating layers, 12 attention heads, 768 hidden layers, 128 embeddings, and 11M parameters. ALBERT reduces the dependency on extremely high computational power making it more feasible for real-life applications.

5 Implementation

This section elaborates upon the details of the implementation of this research project to build a fake review detection model.

5.1 Setup

The below table gives details about the programming languages, technologies, and libraries used throughout the project.

Table 2: Configuration

IDE	Google Colab Pro (Cloud-based)
Computation	GPU
Type	Tesla P100-PCIE-16GB
Number of GPU	1
Programming language	Python
Framework	Pytorch
Modeling library	SimpleTransformer, HuggingFace Transformer, Sklearn, Pandas, Numpy, Matplotlib, Seaborn, Wandb

5.2 Data Loading

The first step carried out in implementation phase is loading the datasets in Python. Both 'Yelp hotel reviews' and 'Yelp restaurant reviews' datasets were available as SQLite Database file with .db extension. To store the SQLite query results into a panda DataFrame, the 'sqlite3' library available in Python is used. First, a connection is set up with

the database using sqlite3.connect function and then using the read_sql_query function available in Pandas library, all the rows present in the 'Review' table of the database are stored in the DataFrame. This step is carried out for both datasets separately. Post which these DataFrames are combined together to create a single DataFrame and converted into CSV format, which is used in the subsequent phases of implementation.

5.3 Data Pre-processing

The CSV file is stored in Google drive. The drive is mounted in Google Colaboratory, and the CSV file is loaded as a pandas Dataframe. First, the dataset is checked for null or missing values. As our data did not consist of missing values, we moved forward to exploratory data analysis. A pie chart is plotted to display the percentage distribution of fake and genuine reviews present in the dataset. Around 59.7% of reviews present in the dataset are genuine reviews, and 40.3% are fake reviews. Next, below mentioned steps are followed to clean the data further.

5.3.1 Handling Contractions

Contraction is the abbreviated term of an English word or a group of words. For example, contraction of I am is I'm. Usually, human-written reviews consist of many such contractions. In order to standardize the reviews before training the model, these contractions are expanded. The contractions library available in Python is employed for this task.

5.3.2 Removing punctuations, numbers, and special characters

The reviews consist of various elements such as punctuations, emojis, special characters, numbers, HTML tags, and undesired white spaces. These elements create unnecessary noise, and these values are removed from the review text to achieve the best performance. To remove numeric characters from reviews, lambda expression and isdigit() method of Python is used. To remove rest of the noises, re library is imported in Python, and using regular expressions (regex), the data is cleaned.

The cleaned review text is stored in the Dataframe in a new column, which is used for further implementation. The dataset is checked for duplicate reviews, and such reviews are eliminated from the Dataframe using drop_duplicates function in Python. This project intends to use transformer-based pre-trained models such as BERT, RoBERTa, DistilBERT, and ALBERT, which are pre-trained on a large corpus of data and use a self-attention mechanism to learn the features. Hence, we have not used traditional text pre-processing techniques such as stop words removal, stemming, and lemmatization to keep the semantic meanings of the reviews intact. The model-specific tokenizers perform tokenization of reviews before model training.

5.3.3 Review length distribution

The pre-trained models require input texts to be of the same length. To achieve this, a maximum length is set depending upon the dataset used. To understand the distribution of review length present in the Yelp dataset, the dist plot is plotted. It is a combination of kdeplot and histogram. In Figure 5 the abscissas represent the length of fake and

genuine reviews, and the ordinates represent probability density function of the kernel. The distribution of both fake and genuine reviews is right-skewed. To minimize the use of computational resources, the mean length of reviews is selected as the maximum length of the input sequence. While training the model, 'max_seq_length' is set as 142.

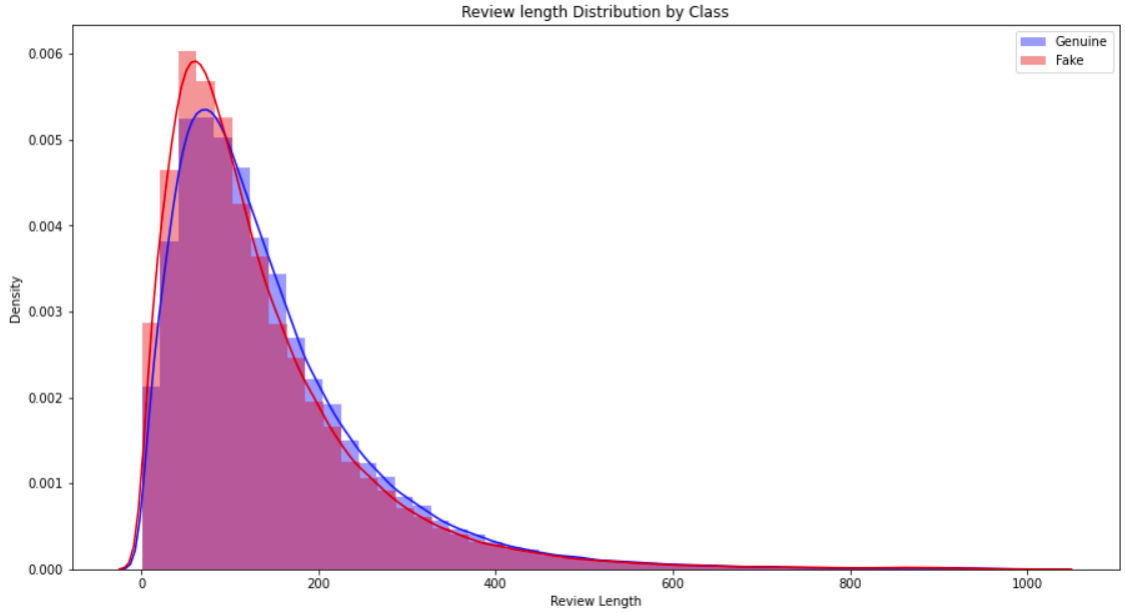

Figure 5: Token length distribution for reviews

5.4 Data Sampling

The Yelp dataset consists of over 1.4 million records labeled as fake and genuine reviews (1 and 0). This project aims to create a state-of-the-art fake review classifier while minimizing the use of computational resources. Using large datasets for training increases the model training time and consequently uses more computational power. Using the sample function in Python, two divisions of the dataset consisting of 10% and 50% data are obtained. All the models are trained on both fractions of dataset, and the performance of models is evaluated. The dataset with 10% of original data has 109700 records and the one with 50% of data has 548502 records. The initial distribution of fake and genuine reviews is maintained in both samples.

5.5 Train-Validate-Test split

The sklearn library, available in Python, is used to split the dataset. Each sample of dataset is segregated into Train-Validate-Test sets using train_test_split function. 80% of data is used for model training and 10% each for evaluation and testing of the built classifier.

5.6 Model training and Evaluation

5.6.1 BERT: Implementation and Results

We have used the Simple Transformers library built on top of Hugging face library for easy implementation of complex transformers models. The ClassificationModel, which is built explicitly for binary and multi-class text classification, is used to build fake review detection classifier. For fine-tuning BERT with the Yelp dataset, bert-base-uncased is used. First, the BERT tokenizer performs tokenization of reviews, and the input text is converted into tokens. The max_seq_length of 142 is given as input to the model training function. The model is trained for two epochs, with training and evaluation batch size as 16. The smaller the batch size, the more is the requirement for computational resources. Hence, we have trained the model using batch size as 32, which is the next recommended batch size by [34]. However, the model's performance was not satisfactory, and hence batch size is fixed as 16. Also, after training the model with different learning rates comparing the performance, the rate 1e-5 is finalized. The model is trained using binary cross-entropy loss function, and to optimize the losses, Adam optimizer with an epsilon value of 1e-8 is used. To prevent the model from overfitting, we have initialized the value of dropout layer as 0.3.

While training, we have also initialized 'wandb_project' parameter. It supports Weights and Biases framework for visualizing model training. The framework creates dashboards that analyze the model's performance and assist in comparing various trained models based on aggregate information. The visualizations provide a clear picture of the utilization of computational resources. Visualization representing running time, parameters used, architecture, and system information is also displayed.

BERT is fine-tuned for both fractions of datasets. 80% of data is used for model training and 10% for evaluating the trained model. Using the predict function, we have tested the fake review classifier on test data after model training. In the ClassificationModel the dense and softmax layers are implicit.

Table 3 displays the performance of BERT for fake review classification based on all evaluation metrics.

Table 3: BERT Results

Evaluation Parameters	**10% Data**	**50% Data**
Accuracy	0.65	0.67
Macro-Precision	0.64	0.66
Macro-Recall	0.64	0.65
Macro-F1 Score	0.64	0.65
Weighted-Precision	0.65	0.67
Weighted -Recall	0.65	0.67
Weighted -F1 Score	0.65	0.67

5.6.2 RoBERTa: Implementation and Results

For fine-tuning RoBERTa with the yelp dataset, roberta-base model is used. RoBERTa was trained using different learning rates and batch sizes. The best performance was achieved for learning rate 1e-5 and batch size 16. For tokenization, RoBERTa uses byte-level Byte-Pair-Encoding tokenizer. The number of epochs for training is 2, and the maximum sequence length is 142. The loss function, optimizer, and other parameters are kept same as BERT. This is to facilitate a fair comparison of different pre-trained models. Table 4 displays the performance of RoBERTa for fake review classification based on accuracy, recall, precision, and F1-score metrics.

Table 4: RoBERTa Results

Evaluation Parameters	**10% Data**	**50% Data**
Accuracy	0.67	0.69
Macro-Precision	0.66	0.68
Macro-Recall	0.66	0.67
Macro-F1 Score	0.66	0.68
Weighted-Precision	0.67	0.69
Weighted -Recall	0.67	0.69
Weighted -F1 Score	0.67	0.69

5.6.3 DistilBERT: Implementation and Results

For fine-tuning DistilBERT with the yelp dataset, the distilbert-base-cased model is used. DistilBERT uses WordPiece tokenizer for tokenization of the input text. The learning rate is considered 1e-5, which yielded optimum results. The number of epoch, dropout layer, maximum sequence length, loss function, optimizer, batch size are kept the same as BERT model for fair comparison of results. Table 5 displays the performance of DistilBERT for fake review classification based on all evaluation metrics.

Table 5: DistilBERT Results

Evaluation Parameters	**10% Data**	**50% Data**
Accuracy	0.65	0.68
Macro-Precision	0.64	0.67
Macro-Recall	0.63	0.66
Macro-F1 Score	0.64	0.66
Weighted-Precision	0.65	0.67
Weighted -Recall	0.65	0.68
Weighted -F1 Score	0.65	0.68

5.6.4 ALBERT: Implementation and Results

For fine-tuning ALBERT with the yelp dataset, albert-base-v1 model is used. It uses SentencePiece tokenizer for tokenization of the input text. The other model parameters

are kept same as the BERT model to facilitate a fair comparison of the performance with other pre-trained models. Table 6 displays the performance of ALBERT for fake review classification based on accuracy, recall, precision, and F1-score metrics.

Table 6: ALBERT Results

Evaluation Parameters	**10% Data**	**50% Data**
Accuracy	0.64	0.66
Macro-Precision	0.62	0.65
Macro-Recall	0.61	0.64
Macro-F1 Score	0.61	0.64
Weighted-Precision	0.63	0.66
Weighted -Recall	0.64	0.66
Weighted -F1 Score	0.63	0.66

6 Discussion

6.1 Comparison of Developed Models based on Performance

We have implemented four pre-trained models, BERT, RoBERTa, DistilBERT, and ALBERT, to build a fake review classifier. To train the classifiers, we have used actual data obtained from Yelp.com containing reviews about multiple domains, making it a generalized dataset. The Yelp filtering algorithm classified these reviews as filtered (fake) and unfiltered (non-fake). As highlighted by Mukherjee et al. (2013), this dataset consists of reviews close to the ground truth written by real spammers to promote or demote businesses on social platforms.

BERT has achieved an accuracy of 67%, along with a weighted F1 score of 0.67, using 50% of records from the dataset for training. This model lacks in performance when compared with some works discussed in the literature. This can be accounted to the fact that for most models, the evaluation is performed on Opspam or a similar dataset that lacks real-life data, and classification ground-truth reviews belonging to various domains is a complex task. Apart from this, the researchers who used variations of different Yelp datasets available have rigorously filtered the dataset to narrow down the data to a specific domain, for example, Restaurants of a particular city (Kennedy et al.; 2020). The fake review classifier developed by us is more generalized as it is trained with data from various domains.

The fake review classifier built by fine-tuning DistilBERT on 50% data has achieved an accuracy of 68% with a weighted F1-score of 0.68. DistilBERT has performed better than BERT to detect fake reviews. BERT and DistilBERT both have outperformed the classifier built by fine-tuning ALBERT, which achieved an accuracy of 66% with weighted F1-score of 0.66 when trained on 50% dataset. Out of all the models built to detect fake reviews, the classifier built by fine-tuning RoBERTa has shown the best performance. It is capable of detecting fake reviews with an accuracy of 69%. The model has also achieved an F1-score of 0.69.

We have also fine-tuned all the models by using just 10% of the dataset. BERT, RoBERTa, DistilBERT, and ALBERT have achieved an accuracy of 65%, 67%, 65%, 64%, and weighted F1-score of 0.65, 0.67, 0.65, 0.63, respectively, when trained on 10% of the dataset. For all the pre-trained models, training with 50% of the dataset has produced better results when compared with models trained using 10% of the dataset. The difference between accuracy lies between 2 to 3% for each model, with the model trained on 50% performing better.

6.2 Comparison of Developed Models based on Computational Time

Figure 6 shows the time taken by models to complete the training. The training time for all applied pre-trained models trained on 10% of dataset and 50% of the dataset is shown in the bar graph. The training time with 10% of data is significantly less than training time with 50% data. BERT takes the least amount of time for 10% of data, followed by DistilBERT, ALBERT, and RoBERTa. When 50% of data is used, then DistilBERT is the fastest, followed by ALBERT. The time taken for training BERT and RoBERTa is identical and significantly higher than DistilBERT.

High computational resources are required for training the models. To minimize the use of computational power, a trade-off between the model's accuracy to detect fake reviews and time needed for training can be considered. Though, RoBERTa trained on 50% data yields the best results, DistilBERT can achieve a comparable result (1% less accuracy) with 43% reduction in training time.

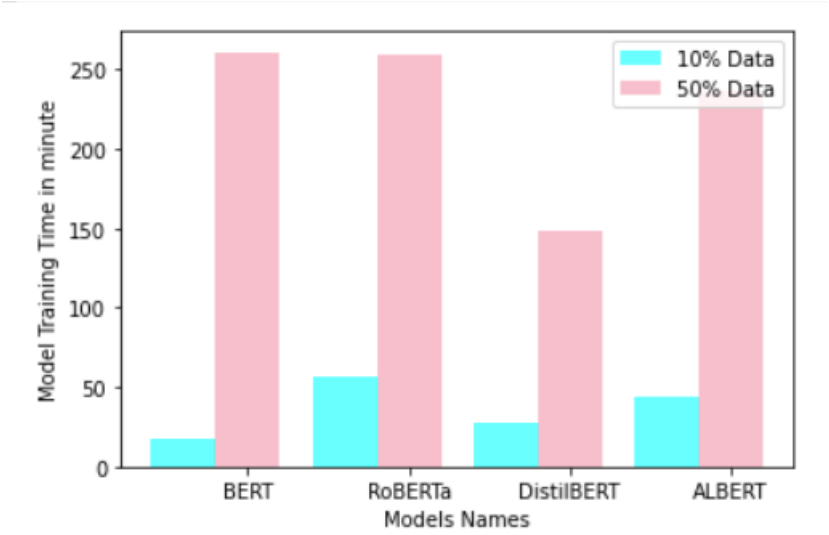

Figure 6: Training time for developed models

6.3 Comparison of Developed Models with Baseline Model

The restaurant domain-specific model built by Mukherjee et al. (2013) is considered baseline for this research work. The authors achieved an accuracy of 67.8% using SVM.

The RoBERTa fake review classifier, trained using 50% data, outperforms the baseline model. It has achieved 69% accuracy and F1-score of 0.69.

7 Conclusion and Future Work

The reviews present on online platforms influence the purchasing decision of future customers and have a financial impact on businesses. Spam reviews written with an agenda mislead the customers, making identification of fake reviews crucial. Researchers have rigorously used machine learning and neural networks to build fake review detection systems. However, the requirement of domain knowledge to perform feature engineering for traditional machine learning algorithms and the need for large labeled datasets for deep learning are few challenges of these models. We have leveraged transfer learning and used transformer-based pre-trained models, BERT, RoBERTa, ALBERT, and DistilBERT, to build fake review classifier. These models are trained using 10 and 50% data of Yelp dataset to construct a generalized fake review classifier while minimizing the use of computational resources. Data pre-processing and hyper-parameters selection are performed, and the pre-trained models are fine-tuned for both samples of data. Performance of all the models is evaluated, considering accuracy and weighted F1-score as the primary metric for evaluation. RoBERTa trained using 50% of data outperforms the baseline model and achieves accuracy of 69% and a weighted F1-score of 0.69. The model's architecture can be modified for future work by adding a BiLSTM layer instead of a dense layer. Also, an ensemble of pre-trained models of the BERT family can be built to achieve a better performing fake review classifier.

References

Alsubari, S. N., Shelke, M. B. and Deshmukh, S. N. (2020). Fake reviews identification based on deep computational linguistic, *International Journal of Advanced Science and Technology* **29**: 3846–3856.

Bhuvaneshwari, P., Rao, A. N. and Robinson, Y. H. (2021). Spam review detection using self attention based cnn and bi-directional lstm, *Multimedia Tools and Applications* **80**(12): 18107–18124.

Bidgolya, A. J. and Rahmaniana, Z. (2020). A robust opinion spam detection method against malicious attackers in social media, *arXiv preprint arXiv:2008.08650* .

Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805* .

Elmogy, A. M., Tariq, U., Mohammed, A. and Ibrahim, A. (2021). Fake reviews detection using supervised machine learning, *Int. J. Adv. Comput. Sci. Appl* **12**.

Guo, Z., Tang, L., Guo, T., Yu, K., Alazab, M. and Shalaginov, A. (2021). Deep graph neural network-based spammer detection under the perspective of heterogeneous cyberspace, *Future Generation Computer Systems* **117**: 205–218.

Hajek, P., Barushka, A. and Munk, M. (2020). Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining, *Neural Computing and Applications* **32**(23): 17259–17274.

Hernández-Castañeda, Á., Calvo, H., Gelbukh, A. and Flores, J. J. G. (2017). Cross-domain deception detection using support vector networks, *Soft Computing* **21**(3): 585–595.

Hu, N., Bose, I., Gao, Y. and Liu, L. (2011). Manipulation in digital word-of-mouth: A reality check for book reviews, *Decision Support Systems* **50**(3): 627–635.

Hu, N., Liu, L. and Sambamurthy, V. (2011). Fraud detection in online consumer reviews, *Decision Support Systems* **50**(3): 614–626.

Jain, N., Kumar, A., Singh, S., Singh, C. and Tripathi, S. (2019). Deceptive reviews detection using deep learning techniques, *International Conference on Applications of Natural Language to Information Systems*, Springer, pp. 79–91.

Jindal, N. and Liu, B. (2008). Opinion spam and analysis, *Proceedings of the 2008 international conference on web search and data mining*, pp. 219–230.

Jnoub, N. and Klas, W. (2020). Declarative programming approach for fake review detection, *2020 15th International Workshop on Semantic and Social Media Adaptation and Personalization (SMA*, IEEE, pp. 1–7.

Kennedy, S., Walsh, N., Sloka, K., Foster, J. and McCarren, A. (2020). Fact or factitious? contextualized opinion spam detection, *arXiv preprint arXiv:2010.15296* .

Kudo, T. and Richardson, J. (2018). Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing, *arXiv preprint arXiv:1808.06226* .

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations, *arXiv preprint arXiv:1909.11942* .

Li, J., Lv, P., Xiao, W., Yang, L. and Zhang, P. (2021). Exploring groups of opinion spam using sentiment analysis guided by nominated topics, *Expert Systems with Applications* **171**: 114585.

Ligthart, A., Catal, C. and Tekinerdogan, B. (2021). Analyzing the effectiveness of semi-supervised learning approaches for opinion spam classification, *Applied Soft Computing* **101**: 107023.

Lim, E.-P., Nguyen, V.-A., Jindal, N., Liu, B. and Lauw, H. W. (2010). Detecting product review spammers using rating behaviors, *Proceedings of the 19th ACM international conference on Information and knowledge management*, pp. 939–948.

Liu, M., Shang, Y., Yue, Q. and Zhou, J. (2020). Detecting fake reviews using multidimensional representations with fine-grained aspects plan, *IEEE Access* **9**: 3765–3773.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach, *arXiv preprint arXiv:1907.11692* .

Mukherjee, A., Venkataraman, V., Liu, B., Glance, N. et al. (2013). Fake review detection: Classification and analysis of real and pseudo reviews, *UIC-CS-03-2013. Technical Report* .

Sam, K. M. and Chatwin, C. (2015). Ontology-based sentiment analysis model of customer reviews for electronic products, *Encyclopedia of Information Science and Technology, Third Edition*, IGI Global, pp. 892–904.

Sanh, V., Debut, L., Chaumond, J. and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, *arXiv preprint arXiv:1910.01108* .

Schuster, M. and Nakajima, K. (2012). Japanese and korean voice search, *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, pp. 5149–5152.

Sennrich, R., Haddow, B. and Birch, A. (2015). Neural machine translation of rare words with subword units, *arXiv preprint arXiv:1508.07909* .

Shan, G., Zhou, L. and Zhang, D. (2021). From conflicts and confusion to doubts: Examining review inconsistency for fake review detection, *Decision Support Systems* **144**: 113513.

Tang, H. and Cao, H. (2020). A review of research on detection of fake commodity reviews, *Journal of Physics: Conference Series*, Vol. 1651, IOP Publishing, p. 012055.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. and Polosukhin, I. (2017). Attention is all you need, *Advances in neural information processing systems*, pp. 5998–6008.

Wang, C.-C., Day, M.-Y., Chen, C.-C. and Liou, J.-W. (2018). Detecting spamming reviews using long short-term memory recurrent neural network framework, *Proceedings of the 2nd International Conference on E-commerce, E-Business and E-Government*, pp. 16–20.

Wang, G., Xie, S., Liu, B. and Yu, P. S. (2012). Identify online store review spammers via social review graph, *ACM Transactions on Intelligent Systems and Technology (TIST)* **3**(4): 1–21.

Wang, J., Kan, H., Meng, F., Mu, Q., Shi, G. and Xiao, X. (2020). Fake review detection based on multiple feature fusion and rolling collaborative training, *IEEE Access* **8**: 182625–182639.