Detecting Fake Online Reviews using Fine-tuned BERT

DAVID REFAELI

Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel, e-mail: DavidRefaeli@gmail.com

PETR HAJEK

Faculty of Economics and Administration, University of Pardubice, Studentska 84, 53210 Pardubice, Czech Republic, e-mail: petr.hajek@upce.cz

Fake online reviews are becoming a major problem nowadays with the growing number of online purchases. Recently, natural language processing (NLP) methods that analyze the content of reviews have been increasingly used to detect fake reviews. The problem becomes extremely difficult due to the lack of reliable data caused by the difficulty in labeling fake and honest reviews. In this paper, we not only conduct a structural taxonomy of this topic, but we also present extensive experiments using a state-of-the-art language model BERT (Bidirectional Encoder Representations from Transformers) on different online review datasets. By efficiently fine-tuning this model, we outperform existing detection models by achieving 91% accuracy on the balanced crowdsourced dataset of hotel, restaurant, and doctor reviews and 73% accuracy on the imbalanced third-party Yelp dataset of restaurant reviews.

CCS CONCEPTS • Information systems → Information systems applications → Decision support systems → Data analytics • Computing methodologies → Artificial intelligence → Natural language processing → Information extraction.

Additional Keywords and Phrases: Fake review, Detection, BERT, Fine-tuning

1 INTRODUCTION

Fake online content is becoming a big problem for humanity in the age of the internet and social media. In fact, the problem of opinion spamming goes beyond mere online reviews. Political parties may use fake content in the form of fake news and fake comments posted on news and social media to influence public opinion [17]. Opinion spamming through fake reviews remains one of the main areas of research interest. Fake reviews are reviews pretending to be honest, although they are motivated by external motives (e.g., financial gain) and aim to promote or to demote a product or company. It has been reported that up to 90% consumers read online reviews before making a purchase decision and most of these consumers trust the authenticity of the comments [12]. However, it is also reported that the proportion of fake reviews is steadily increasing [8].

There are several methods proposed for detecting fake online reviews. A somewhat new and widely discussed method is based solely on intrinsic evaluation [18], in which a review is judged solely on grounds of its content, with no additional information about the review and the reviewer's behaviour. The underlying assumption is that fake writing is inherently different from honest writing. The theoretical justification is drawn from psycholinguistic studies on deception and on "reality monitoring", according to which deceptive writing is characterized, for example, by psychological distancing, difficulties in encoding spatial information, and therefore encodes identifiable cues [14].

Other methods use (comparative behavioral) extrinsic evaluation to detect fake reviews, fake reviewers, and groups of reviewers [16]. These methods essentially look for suspicious behaviors (e.g., inappropriately similar reviews across products or across reviewers; a disproportionate number of reviews per day per product or per reviewer; anomalous distribution of the number of reviews per product or per reviewer; and synchronized behavior across groups of reviewers). These approaches include anomaly detection [8], time series analysis [25], and graph-based methods [20] and rely on the metadata about reviews and reviewers.

Research on this issue appears to have been primarily divided along this main line, although the division is not strict, and some studies use a combination of both approaches.

Online consumer companies like Amazon and Yelp are implementing their own (proprietary) algorithms that allow to filter out fake reviews. It is speculated that they rely heavily on extrinsic evaluation [16]. It is also argued (ibid) that these companies do a relatively good (though not perfect) job in filtering out fake reviews.

Humans are particularly bad at detecting fake reviews based on intrinsic evaluation, with research showing that their accuracy tends to be even lower than that of random selection [17,18]. Machine learning algorithms seem to significantly outperform humans, achieving approximately 90% accuracy on benchmark datasets [5]. Although reasons suggest that humans could perform decently on extrinsic evaluations, this is not practically feasible due to information overload, so machine learning algorithms are preferred.

Of the machine learning models used so far, the BERT-based pre-trained models fine-tuned for the task of detecting fake reviews reportedly achieved the most promising results in terms of accuracy [7,22]. BERT outperformed other machine learning models in [7], showing the effectiveness of contextualized embeddings compared to sparse bag-of-words representations other dense representations (non-contextual pre-trained word2vec embeddings). In [22] BERT outperformed RNN's and in some cases also CNN's architectures used, showing that the context-awareness of words in both text directions outperforms traditional context-free dense models. However, these related studies relied on the baseline BERT model without investigating different fine-tuning methods. Recent empirical evidence suggests that superior performance can be achieved in a variety of text classification tasks using an appropriate fine-tuning process [24]. Inspired by these findings, here we conduct extensive experiments to investigate the effects of different fine-tuning methods on the detection of fake online reviews using BERT.

The remainder of this paper is organised as follows. The next section reviews related work on detecting fake online reviews. Section 3 presents the datasets used and the experimental setup of the BERT model. Section 4 presents our results, and Section 5 concludes with potential future work.

2 RELATED WORK

Table 1 provides an overview of related work and their results. Here we focus on the NLP domain.

As noted above, the problem of labelled data is acute in this field. It is not easy to label reviews as fake or honest. Several methods have been employed so far. The duplicity / content similarity method was used in [6] to create a dataset by labelling duplicates or near-duplicates reviews as fake. Lau et al. [9] used a similar approach, combining unsupervised (cosine-similarity) search to detect suspicious reviews, and then labelling them by hand. Only untruthful reviews with at least one or more word substitutions were included. The anomaly / suspicious behaviour approach was used by Mukherjee et al. [15] by first identifying suspicious "groups" (through extrinsic comparative evaluations), and then employing a team of eight human experts to manually score them. Similarly, Feng et al. [3] labelled fake reviews according to their distributional anomaly. The crowdsourcing approach was first used by Ott et al. [18,19]. This was a major contribution by creating a public "gold" labelled dataset, paying anonymous Amazon-Mechanical-Turk workers to generate fake hotel reviews, and collecting real reviews of the same hotels on TripAdvisor. The same methodology was later applied to the reviews of restaurants and doctors [10]. A similar approach was used by Oh et al. [17] to collect 866 truthful and 869 deceptive comments on social issues. Other studies relied on third parties, mainly consumer platforms, to label reviews, assuming they had the competence to do so. The Amazon dataset [5] contains a sample of 21,000 reviews of different products, half of which were labelled as "non-compliant" by Amazon's algorithm. The Yelp dataset [16] contains nearly a million restaurant reviews from different areas in the US. Martens et al. [13] used impressive investigation techniques to

label fake app reviews. They “hacked” fake reviews service providers by (1) soliciting them for examples (feigning interest in the service), (2) signing up to these platforms and extracting lists of apps requesting fake reviews, and (3) in the case of one provider, finding a cache of screenshots of unrestricted reviews and converting them to text using OCR technology. None of these methods of labelling data is perfect. They all suffer from different problems. The duplicity methods have been criticized for not being sophisticated enough because modern spammers could be more careful to avoid duplication. Metadata used to indicate anomalies may not be available in in some cases. Some reviews are anonymous by design (e.g., Glassdoor reviews). Crowdsourcing data has been criticized for not truthfully representing fake reviews "in the wild", having too different language distribution, making them too easily identifiable by intrinsic methods. Third party labelling may not be completely accurate and, therefore, cannot be regarded as ground truth. Fake reviews revealed by the investigation method are undoubtedly fake, though there is no guarantee that those that have not been revealed are truly honest (and the same goes for reviews labelled as "honest" by the duplicity and anomaly methods).

NLP-based models use features extracted from the content of reviews. These features can be used for both intrinsic and comparative/extrinsic evaluations. For example, stylistic features are extracted from the text only, although they are used comparatively to detect the spamming author based on his/her style [23]. Other features include:

- Discrete / sparse representations: these are bag-of-words representations of content, such as an *n*-grams (unigrams, bigrams, trigrams) [18,19]. The representation can use the words themselves or their syntactic structures (e.g., POS tagging, POS emissions) or both [4]. They can be binary (0,1) or counted (# of occurrences), and can be smoothed (e.g., Knesser-Ney) or transformed (e.g., TF-IDF).
- Engineered and metadata features: these are heuristically extracted features such as review length (character-based, token/word-based) [6], average token length, percentage of words written in capitals, proportion of first-person pronouns, etc. (for further examples, see the respective tables in [23] and [20]). Another example is LIWC (Linguistic Inquiry and Word Count) [16], which counts and groups the number of occurrences of nearly 4,500 keywords into 80 psychologically meaningful dimensions.
- Distributed / dense representations: these are representations obtained using deep neural networks (DNNs), including Word2Vec, convolutional NNs (CNNs), recurrent NNs (RNNs) such as vanilla, gated recurrent units (GRU), and long short-term memory (LSTM; unidirectional and bidirectional), and transformers (BERT, GPT-2, RoBERTa, etc.).

These features are then fed into a classifier that determines whether a review is fake or honest. Table 1 shows that for the crowdsourcing datasets, the intrinsic models trained and evaluated on a specific domain (hotels, restaurants, doctors) achieve almost 90% accuracy regardless of their feature representation. However, the intrinsic methods do not seem to work as well on the Yelp dataset, where they achieve only 68.5% accuracy only. Previous studies [16] suggested that crowdsource data is different than reviews “in the wild”, though other explanations are possible, e.g., that third-party labelling concerns itself with other criteria than just the honesty of the review (such as the quality of the review in general). Overall, it seems that the intrinsic methods are very data-specific, as they need to be trained on each domain separately to achieve good results.

Table 1: List of related studies

Study	Data	Labeling	Domain	Intrinsic/Extrinsic	Features	Model	Acc	F1	AUC
[6]	Amazon	Duplicity	Mixed	Mixed	B	LR	NA	NA	0.78
[18]	Hotels	Crowdsourcing	NLP	Intrinsic	A+B	SVM	0.898	0.898	NA
[9]	Amazon	Duplicity	NLP	Extrinsic	A	Unsup.	NA	NA	0.998
[4]	Hotels	Crowdsourcing	NLP	Intrinsic	A	SVM	0.912	NA	NA
	Yelp	3rd party					0.643	NA	NA
[15]	Amazon	3rd party	non-NLP	Extrinsic	B	GSRank	NA	NA	0.95
[16]	Yelp	3rd party	NLP	Intrinsic	A+B	SVM	0.685	0.721	NA
			Mixed				0.861	0.857	NA
[19]	Hotels	Crowdsourcing	NLP	Intrinsic	A+B	SVM	0.86	0.86	NA
[23]	Hotels	Crowdsourcing	NLP	Extrinsic	B	SVM	NA	0.84	NA
[10]	Restaurants	Crowdsourcing	NLP	Intrinsic	A+B	SVM	0.785	0.778	NA
	Doctors	Crowdsourcing				SAGE	0.647	0.628	NA
[20]	Yelp	3rd party	Mixed	Extrinsic	B	SpEagle	NA	NA	0.794
[1]	Hotels	Crowdsourcing	NLP	Intrinsic	A	SVM	0.90	NA	NA
	News	3rd party					0.92	NA	NA
[11]	Hotels+	Crowdsourcing	NLP	Intrinsic	C	CNN	0.80	0.834	NA
[21]	Hotels+	Crowdsourcing	NLP	Intrinsic	C	CNN	0.759	0.74	NA
						GRU	0.836	0.834	NA
[8]	Yelp	3rd party	non-NLP	Extrinsic	B	GMM	NA	NA	0.70
[13]	Apps	Investigation	Mixed	Extrinsic	B	RF	0.97	0.97	0.989
[7]	Hotels	Crowdsourcing	NLP	Intrinsic	C	BERT	0.891	NA	NA
[5]	Hotels	Crowdsourcing	NLP	Intrinsic	C	NN	0.895	0.869	0.951
	Restaurants	Crowdsourcing				CNN	0.898	0.90	0.956
	Doctors	Crowdsourcing				CNN	0.883	0.91	0.946
	Amazon	3rd party				NN	0.828	0.825	0.893
[17]	Comments	Crowdsourcing	NLP	Intrinsic	A	NN	0.812	0.809	NA
[22]	Yelp	3rd party	NLP	Intrinsic	C	BERT	NA	0.69	NA
This study	Hotels+	Crowdsourcing	NLP	Intrinsic	C	BERT	0.91	0.93	NA
	Yelp	3rd party					0.73	0.73	NA

Notes: only the best results are presented. Hotels+ refers to the crowdsourcing data collected by [10,18,19], including reviews of restaurants and doctors. Features: A = sparse, B = engineered, C = dense. Acc is accuracy, AUC is area under a receiver operating characteristic curve, F1 is F1-score, GMM – Gaussian mixture model, LR – logistic regression, RF – random forest, SVM – support vector machine, SAGE – sparse additive generative model, Unsuper. – unsupervised learning.

3 DATA AND METHODS

In this study, we experimented with two different datasets, one labelled using crowdsourcing and the other labelled by a third party.

Crowdsourced (CS) reviews[1] – the Hotel dataset [18,19] consists of 1,880 negative and positive reviews (half and half). Of these, 800 fake reviews were generated by Amazon-Mechanical-Turk (AMT), 280 fake reviews were generated by "experts", and 800 "real" reviews were collected from TripAdvisor. The Restaurant dataset consists of 400 positive reviews (200 fake, 200 "real"), and the Doctor dataset consists of 556 positive reviews (356 fake, 200 "real").

Yelp reviews[2] – a collection of restaurant and hotel reviews consists of both 67k reviews of Chicago establishments used by [16] and another ~1m reviews collected by [20]. Their labels are based on Yelp's proprietary algorithm and are considered "near" ground truth. We used only a subset of the Rayana [20] dataset (ZIP). The Yelp dataset is unique in that it also contains extensive user metadata that allows for more comparative analyses (though we did not use them in this study because we focused on intrinsic evaluation).

Our experiments were conducted using the BERT language model. BERT [2] is a novel Transformer-based model (based on a multi-layer bidirectional Transformer) that utilizes pre-training (on masked language model and next sentence prediction tasks) and can be used for fine-tuning various NLP tasks, among which text classification. All runs were performed on the BERT-base (the base model with 12 Transformer blocks, 12 self-attention heads, and the hidden size of 768), using the AdamW optimizer (with default betas). We tested cased vs. uncased (text was lowercased before tokenization), maximum sequence length of 256 vs. 512 (maximum), different learning rates (with and without a scheduler, to avoid the overfitting problem), different levels of dropout regularization (0.1, 0.3, 0.5), different numbers of training epochs, and two different classifier models (without and with freezing BERT layers, which can be beneficial for small datasets as it usually reduces the risk of overfitting).

4 EXPERIMENTAL RESULTS

The maximum sequence length of the BERT model is 512 tokens. If the value is exceeded, the input text is truncated. BERT uses its own tokenizer, which also splits some words to sub-words and, in some case, into characters. Most of the reviews were between 40 and 240 words, so we decided to run on a subset of reviews of that length as well, using a maximum sequence length of 256.

We started with the CS review data. To investigate the effect of BERT parameters, we split the data into training, validation and test data in a 80:10:10 ratio. Freezing the BERT weights and adding a 2-layer NN classifier on top of that (projecting the 768 dimensions to 512, and then to 2) gave the highest accuracy of 76% (Table 2). We therefore abandoned this architecture and instead tuned BERT using the standard Bert-For-Sequence classifier from the hugging-face transformers module (which adds a dropout and a 1-layer NN projecting 768 nodes directly to 2, on top of BERT). This in turn allowed us to increase the accuracy to 91% (Table 2).

The CS data contains 240 fake hotel reviews written by "experts", and it is suggested that these reviews are considerably different from the AMT fake reviews, so we also ran the experiments without them, which seems to have actually helped improve performance somewhat (Table 3). We performed them on the entire mix of the CS data and achieved 89% accuracy with five-fold cross-validation, which to our knowledge is the highest reported accuracy on this mix. The highest accuracy in a single run was 91%.

[1] https://myleott.com/op-spam.html

[2] http://odds.cs.stonybrook.edu/yelpzip-dataset/

Table 2: Performance of BERT on the CS dataset

Architecture	Max. length	Frozen	Learning rate	Scheduler	Dropout	Epochs	Acc	F1	Train time
BERT1	512	Yes	1.00E-04	No	0.1	10	0.76	0.81	~10 min
BERT1	256	Yes	5.00E-05	No	0.1	4	0.73	0.78	~10 min
BERT2	512	No	5.00E-05	Linear	0.1	4	**0.91**	**0.93**	~14 min
BERT2	256	No	5.00E-05	Linear	0.1	2	0.85	0.89	~1 min

Notes: BERT1 denotes BERT-base + 2-layer NN, BERT2 is BERT-base + dropout and 1-layer NN. Only the best results are presented for each architecture due to space constraints.

Table 3: 5-fold performance of BERT on the CS dataset without "expert" fake reviews

	Uncased		Cased	
Data	Max length = 256	Max length = 512	Max length = 256	Max length = 512
with "expert" fake reviews	0.86	0.88	0.87	0.88
without "expert" fake reviews	**0.89**	0.87	0.88	**0.89**

Notes: BERT was fine-tuned with a dropout layer (dropout rate = 0.1), learning rate of 5.00E-05 (for the max length of 256) and 2.00E-05 (for 512), 4 epochs, and linear scheduler.

We trained the same model on a subset of the Yelp dataset (Yelp-ZIP), with "honest" reviews subsampled to achieve an even distribution. We were able to obtain results with an accuracy of 73% (Table 4), with the best result obtained for the largest sample of ~76k reviews (all reviews containing 60-220 words).

Table 4: Performance of BERT on the Yelp dataset

Data	Max. length	Frozen	Learning rate	Scheduler	Dropout	Epochs	Acc	F1	Train time
20K sample	256	No	5.00E-06	No	0.1	4	0.69	0.65	~40 min
76K sample	256	No	5.00E-06	No	0.1	3	**0.73**	**0.73**	~40 min

Notes: Only the best results are presented for each sampled dataset due to space constraints.

Some general findings from our experiments are as follows:

- There was no noticeable difference between the uncased and cased version of BERT, suggesting that BERT does not concern itself with text style when detecting fake reviews.
- Using the BERT outputs unaltered (freezing the weights in layers) and training the classifier on top of this model had worse results than fine-tuning the entire BERT model. Therefore, the domain-specific fine-tuning of BERT layers is recommended for the detection of fake reviews.
- Training only 2-4 epochs is sufficient. Beyond that, the model seems to overfit. Increasing the dropout rate does not seem to help either. When fine-tuning BERT for this task, we therefore recommend using a maximum number of epochs of 4 and the dropout rate of 0.1.
- By efficiently fine-tuning BERT, we achieved state-of-the-art performance for both datasets (Table 5).

Table 5: Comparison of BERT performance with state-of-the-art models

Study	Data	Model	Acc	F1
[11]	Hotels+	CNN	0.80	0.83
[21]	Hotels+	CNN	0.76	0.74
		GRU	0.84	0.83
This study	Hotels+	BERT	**0.91**	**0.93**
[4]	Yelp	SVM	0.64	NA
[13]	Yelp	SVM	0.69	0.72
[22]	Yelp	BERT	NA	0.69
This study	Yelp	BERT	**0.73**	**0.73**

5 CONLCUSION

Detecting fake reviews remains a challenging and unresolved problem. Nevertheless, this study appears to provide support for the conclusion that intrinsic methods using only the content of reviews, could be effectively used as part of the solution.

We here performed a structural taxonomy of the topic, touching on various major research methods, data labelling methods, features and classifiers that have been used so far for this problem. We also performed extensive experiments with the cutting-edge context-aware dense language model. While it is still not clear what role the different content representations play, this study used a distributed representation derived by the new context-aware BERT to achieve 91% accuracy on a mixture of crowdsourced reviews and 73% accuracy on a third-party Yelp dataset, which are, to the best of our knowledge, the highest known accuracies. Further research is therefore recommended in the area of BERT model (and other context-aware language models) validation and fine-tuning. Specifically, different BERT layers should be investigated to better fit the model to the target task of fake review detection. In-domain or cross-domain pre-training [24] is another vital issue for future research.

ACKNOWLEDGMENTS

This article was supported by the scientific research project of the Czech Sciences Foundation Grant No. 19-15498S. We also thank Dr. Jiwei Li and Dr. Shebuti Rayana for providing us with the datasets.

REFERENCES

[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. 2018. "Detecting opinion spams and fake news using text classification." Security and Privacy 1(1), e9, 1-15.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. "BERT: Pre-training of deep bidirectional transformers for language understanding". In Proceedings of North American Chapter of the Association for Computational Linguistics, NAACL, 4171-4186.

[3] Song Feng, Longfei Xing, Anupam Gogar, and Yejin Choi. 2012a. "Distributional footprints of deceptive product reviews. " In Proceedings of the International AAAI Conference on Web and Social Media, 6(1), AAAI, 98-105.

[4] Song Feng, Ritwik Banerjee, and Yejin Choi. 2012b. "Syntactic stylometry for deception detection." In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, ACL, 171-175.

[5] Petr Hajek, Aliaksandr Barushka, and Michal Munk. 2020. "Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining." Neural Computing and Applications 32(23), 17259-17274.

[6] Nitin Jindal and Bing Liu. 2008. "Opinion spam and analysis." In Proceedings of the 2008 International Conference on Web Search and Data Mining, ACM, 219-230.

[7] Stefan Kennedy, Niall Walsh, Kirils Sloka, Andrew McCarren, and Jennifer Foster. 2019. "Fact or factitious? Contextualized opinion spam detection." Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, ACL, 344-350.

[8] Naveen Kumar, Deepak Venugopal, Liangfei Qiu, L., and Subodha Kumar. 2019. "Detecting anomalous online reviewers: An unsupervised approach using mixture models." Journal of Management Information Systems 36(4), 1313-1346.

[9] Raymond Y. Lau, S. Y. Liao, Ron Chi-Wai Kwok, Kaiquan Xu, Yunqing Xia, and Yuefeng Li (2012). "Text mining and probabilistic language modeling for online review spam detection." ACM Transactions on Management Information Systems (TMIS) 2(4), 1-30.

[10] Jiwei Li, Myle Ott, Claire Cardie, and Eduard Hovy. 2014. "Towards a general rule for identifying deceptive opinion spam." In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 1, ACL, 1566-1576.

[11] Luyang Li, Bing Qin, Wenjing Ren, and Ting Liu. 2017. "Document representation and feature combination for deceptive spam review detection". Neurocomputing 254, 33-41.

[12] Ying Lin. 2020. 10 Online Review Statistics You Need to Know in 2021. Retrieved July 10, 2021 from https://www.oberlo.com/blog/online-review-statistics

[13] Daniels Martens and Walid Maalej. 2019. "Towards understanding and detecting fake reviews in app stores." Empirical Software Engineering 24, 3316-3355.

[14] Jaume Masip, Siegfried L. Sporer, Eugenio Garrido, and Carmen Herrero. 2005. "The detection of deception with the reality monitoring approach: A review of the empirical evidence." Psychology, Crime & Law 11(1), 99-122.

[15] Arjun Mukherjee, Bing Liu, and Natalie Glance. 2012. "Spotting fake reviewer groups in consumer reviews." In Proceedings of the 21st International Conference on World Wide Web, 191-200.

[16] Arjun Mukherjee, Vivek Venkataraman, Bing Liu, and Natalie Glance. 2013. “What yelp fake review filter might be doing?." In 7th International AAAI Conference on Weblogs and Social Media, AAAI, 409-418.

[17] Yu Won Oh and Chong Hyun Park. 2021. "Machine cleaning of online opinion spam: developing a machine-learning algorithm for detecting deceptive comments." American Behavioral Scientist 65(2), 389-403.

[18] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. 2011. "Finding deceptive opinion spam by any stretch of the imagination." In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, ACL, 309-319.

[19] Myle Ott, Claire Cardie, and Jeffrey T. Hancock. 2013. "Negative deceptive opinion spam." In 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, ACL, 497-501.

[20] Shebuti Rayana and Leman Akoglu. 2015. "Collective opinion spam detection: Bridging review networks and metadata." In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 985-994.

[21] Yafeng Ren and Donghong Ji. 2017. "Neural networks for deceptive opinion spam detection: An empirical study." Information Sciences 385, 213-224.

[22] Yassien Shaalan, Xiuzhen Zhang, Jeffrey Chan, and Mahsa Salehi. 2021. "Detecting singleton spams in reviews via learning deep anomalous temporal aspect-sentiment patterns". Data Mining and Knowledge Discovery 35, 450-504.

[23] Somayeh Shojaee, Masrah A. A. Murad, Azreen B. Azman, Nurfadhlina M. Sharef, and Samaneh Nadali. 2013. "Detecting deceptive reviews using lexical and syntactic features." In 13th International Conference on Intelligent Systems Design and Applications, IEEE, 53-58.

[24] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. "How to fine-tune BERT for text classification?." In China National Conference on Chinese Computational Linguistics, Springer, 194-206.

[25] Junting Ye, Santhosh Kumar, and Leman Akoglu. 2016. "Temporal opinion spam detection by multivariate indicative signals." In 10th International AAAI Conference on Web and Social Media, AAAI, 743-746.