Robust Fake Review Detection using Uncertainty-Aware LSTM and BERT

Sarah Zabeen*
School of Data and Sciences
Brac University
Dhaka, Bangladesh
sarah.zabeen@g.bracu.ac.bd

Alina Hasan*
School of Data and Sciences
Brac University
Dhaka, Bangladesh
alina.hasan@g.bracu.ac.bd

Md. Farhadul Islam
School of Data and Sciences
Brac University
Dhaka, Bangladesh
md.farhadul.islam@g.bracu.ac.bd

Md Sabbir Hossain
School of Data and Sciences
Brac University
Dhaka, Bangladesh
md.sabbir.hossain1@g.bracu.ac.bd

Annajiat Alim Rasel
School of Data and Sciences
Brac University
Dhaka, Bangladesh
annajiat@gmail.com

***Abstract*—In a web-based world driven by e-commerce, customers are quick to turn to online shopping services. However, the products available for purchase cannot be personally inspected so buyers turn to online product reviews. Potential consumers trust these reviews and are likely to spend more at stores with good evaluations. Sellers are well aware of this phenomenon and are not averse to using such unethical methods to boost the reputation of their own products or plunge that of a rival. In other words, anyone is capable of faking an opinion, which can heavily affect consumer purchasing decisions and business profits. It is not uncommon for companies to hire people to post fake reviews online. Scammers and competitors may even deploy bots to flood review sections with spam and mislead consumers into buying unreliable products. In response, many fake review detection models have been extensively explored in the last decade yet there is still a lack of surveys that analyze and summarize these approaches. Our research addresses this gap by advancing the field of fake review detection. We employ the BERT and LSTM models coupled with the Monte Carlo Dropout technique, on the Yelp Labelled Dataset comprising 10,000 hotel reviews from North America. Our study yields an accuracy of 91.75% using the MCD-embedded BERT model, which outperforms LSTM.**

***Index Terms*—Fake Review Detection, Machine Learning, Monte Carlo Dropout, Uncertainty-Aware, NLP, LSTM, BERT, Robust, Reliability**

I. Introduction

Customers can now express their opinions and reviews on many websites in the era of the internet. These reviews can benefit both organizations and potential customers by providing insight into products or services before making a purchase. There has been a considerable increase in the number of consumer evaluations recently. Such assessments have a substantial impact on the choices of potential buyers. Essentially, customer reviews on social media platforms significantly impact their decision to purchase or not, making such reviews a valuable service for individuals.

Reviews authored by individuals inexperienced with the product or service are known as fake reviews. Consequently, an individual who posts these fake reviews is a spammer. A group of collaborating spammers with a common objective is a collective of spammers.

Many algorithms have been explored for fake review detection, Monte Carlo Dropout (MCD) is one such algorithm employed for uncertainty analysis in risk-free review evaluation. It uses probabilistic Bayesian models which allows the neural network to generate multiple predictions for each instance during testing. By averaging the softmax output of the predictions for each class, MCD can provide a measure of the uncertainty in the predictions of a model.

II. Literature Review

Fake reviews are typically recognized using NLP approaches that prioritize textual data and lexical features, like keywords, n-grams, and linguistic style indicators [1]–[3]. Relevant non-textual features, like the user ID and their location, the number of reviews, and suspicious behaviors, are also considered [4]. Features can comprise both textual and non-textual features as discussed by [5]. Their combination generally improves detection, as evidenced in classification

*These authors contributed equally to this work.

tasks [6]. In contrast to heuristic and behavior-based approaches, the methodology proposes an alternate strategy for detecting fake reviews.

Fake news can be classified using social context features retrieved from Twitter data using specific terms or based on news content [7]. In addition, users retweeting, the time difference between retweets, the retweet rate, and user comments provide important social context and text content features. Another research had several fake news classification models that rely on news content including the text-CNN, HAN, RST, and LIWC compared. Some models only depend on social context including the HPA-BLSTM, while others which include the CSI and TCNN-URG utilize the news content alongside. The study proposed a better-performing model called dEFEND, which includes a prediction component, a co-attention layer, and multiple encoders. [8] found that the model performed better than other models. It achieved an accuracy of 80.8% and an F1 score of 0.755 on the GossipCop dataset. On the PolitiFact dataset, an accuracy of 90.4% and an F1 score of 0.928 was attained. The authors also noted that the removal of either the user comments or the co-attention for news content resulted in a drop in accuracy, indicating that user comments are essential in guiding fake news detection in dEFEND. [9] proposed the use of SVM as the classifier unit and both RNN and GRU for user comments, sentence, and word encoding. The model achieved an accuracy of 80.2% and an F1 score of 0.762 on the GossipCop dataset. On the PolitiFact dataset, an accuracy of 91.2% and an F1 score of 0.932 was attained. However, it also relied on user comments.

BERT has gained significant attention for fake news classification. One study proposed using three BERT models for metadata, justifications, and statements on the LIAR PLUS dataset, achieving an accuracy of 74%. Similarly, a double-BERT model achieved an accuracy of 72% on the LIAR dataset [10]. In a related study involving fake review detection, sentence embeddings were generated by combining word embeddings to improve the classification models' understanding of word relationships, resulting in an increase in accuracy from 80% to 87% using the SVM classifier [11]. For improved learning, one method combined the BERT model with three parallel 1d-CNN blocks with changing kernel-size convolutional layers. This framework outperformed previously established deep learning models, demonstrating the importance of output features from BERT [12].

Additionally, in a sarcasm detection study, BERT was compared to deep learning models using GloVe embeddings, and BERT outperformed, demonstrating its superior ability to learn contextual aspects from data [13].

A research compared the accuracy of multiple machine learning algorithms which attempted to detect fake reviews from the Yelp dataset on restaurant reviews. Authors Elmogy et al. experimented with KNN (K=7), Naive Bayes, SVM, logistic regression, and random forest architectures incorporated with bi-gram and tri-gram language models. The F1 score generated by the KNN architecture outperformed the rest, at 82.3% without behavioral analysis input and 86.3% with it. Therefore the study concluded that feature engineering based on the reviewers' profiles would greatly enhance the performance of the system [14].

Uncertainty-aware models in text classification tasks are gaining popularity, considering the risk factors of misclassification and reliability. Research [15] shows the uncertainty of Transformer-based models such as BERT and XLNet is compared to that of RNN variations such as LSTM and GRU using MCD. BERT outperforms the others in the experiment. In another work, study [16] investigates the applicability of uncertainty estimates based on MCD. The sequence of tests conducted on tasks related to understanding natural language demonstrates that the resulting uncertainty estimates improve the quality of detecting instances that are prone to errors.

III. Research Methodology

A. Classification Model

1) LSTM: In Deep Learning, a variant of the Recurrent Neural Network (RNN) is Long Short-Term Memory networks (LSTMs). The given classification model excels at learning long-term dependencies, making them particularly effective for sequence prediction problems. Furthermore, LSTMs leverage feedback connections in order to process entire sequences of data. Their ability to handle sequential data has led to successful applications in areas such as speech recognition and machine translation. As a special type of RNN, LSTMs have shown exceptional performance across a wide range of problems. LSTMs are divided into four different gates which serve varying purposes. The first is Forget Gate (f), which is used to combine the given input and previous output to ascertain how much of the previous state needs to be preserved, multiplied by the previous state. Secondly, there is the Input Gate (i), which decides what new information is entered into the LSTM state. Here, the output is multiplied with new values and added to the previous state. The Input Modulation Gate (g) has the job of modulating data in the Input Gate by adding non-linearity and making it zero-mean for quicker convergence. Lastly, the Output Gate (o) uses the gated input and the gated previous state to generate a scaling fraction which is then combined with the tanh block output before finally being fed back into the LSTM block.

The LSTM model in our study is depicted in Figure 1. The model has small weight parameters of a total of 194,818.

This is an intentional choice for comparative purposes with our BERT model.

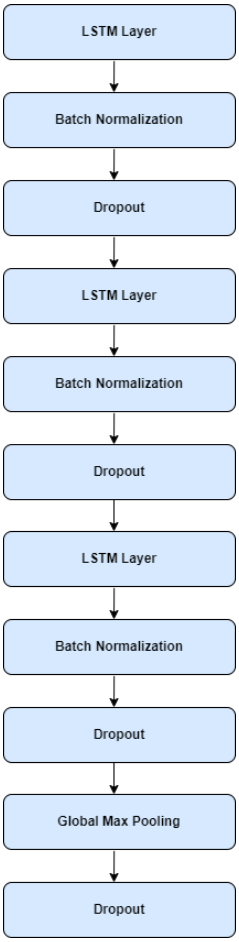

Fig. 1. LSTM Model Structure

2) BERT: The Bidirectional Encoder Representations from Transformers or, BERT is a powerful pre-trained language model that uses a transformer-based bidirectional network architecture. It only uses the encoder part of the transformer to read input and understand the context of words in a text. The model deconstructs the input into individual tokens by transforming them into vector representations which are subsequently fed into the neural network for further processing.

Figure 2 depicts our BERT model. The weight parameters are significantly larger than that of the LSTM model, specifically 109,724,579.

B. Uncertainty Analysis Method: Monte Carlo Dropout

Gal and Ghahramani [17] were the first to suggest Monte Carlo Dropout. They used it in evaluate deep Gaussian processes by employing probabilistic Bayesian models. MCD can produce a series of predictions that depict uncertainty estimations of the experiment at hand. The MCD technique involves completing a number of stochastic forward passes in a Neural Network while also employing active dropout throughout the test stage.

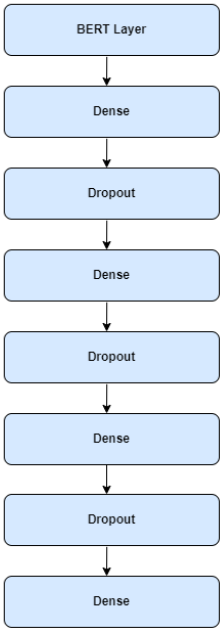

Fig. 2. BERT Model Structure

The process of training a neural network model with the dropout f_{nn} can possibly estimate uncertainty for a given sample, x. This is done by accumulating all the forecasts of T interpretations which utilize numerous dropout masks. In particular, we observe a model with a dropout mask, d_i represented by $f_{nn}^{d_i}$. The subsequent equations demonstrate how the model produces results for a specific sample x:

$$f_{nn}^{d_0}(x), ..., f_{nn}^{d_T}(x) \tag{1}$$

An ensemble prediction can be obtained by calculating the mean and standard deviation. Here, the prediction is the sample average of the posterior probability distribution of the model, which estimates the model's uncertainty regarding x.

$$Predictive\, Posterior\, Mean,\, p = \frac{1}{T}\sum_{i=0}^{T} f_{nn}^{d_i}(x) \tag{2}$$

$$Uncertainty,\ c = \frac{1}{T}\sum_{i=0}^{T}[f_{nn}^{d_i}(x) - p]^2 \tag{3}$$

No changes are made to the dropout NN but the results of the stochastic forward passes are recorded. The predictive mean and model uncertainty is assessed with this method so that the information can be used with pre-existing dropout-trained NN models. This is done for overall uncertainty estimation and finding the list of the most uncertain samples. In this work, we simply change the regular dropouts to MCD. The dropout rate remains the same for both models and both cases.

IV. Experimental Analysis

A. Dataset

The 'Yelp Labelled Dataset: Spam Reviews for New York City' is a valuable resource consisting of 10,000 reviews that were extracted using the official Yelp API, a contribution of Abid Meraj [18]. This dataset contains hotel reviews from North America.

Each review is labeled with one of five possible star ratings ranging from one to five stars, which is a significant feature. Another distinguishing aspect is the inclusion of a label value of 1 when a review is genuine and -1 when it is fake. This addition provides a more comprehensive view of customer feedback on Yelp and can be particularly useful for analyzing fake reviews and creating detection algorithms.

B. Experimental Setup

The models are trained using NVIDIA 3080 Ti and the model development is done using the Tensorflow-Keras framework. The batch size is 8 and the dropout rate is 25% for both models. LSTM requires 10 epochs while BERT requires 3. The evaluation metrics use accuracy, F1-Score, and binary cross-entropy loss. Both models make use of the Adam optimizer, which has a learning rate of 2×10^{-5}.

C. Performance Analysis

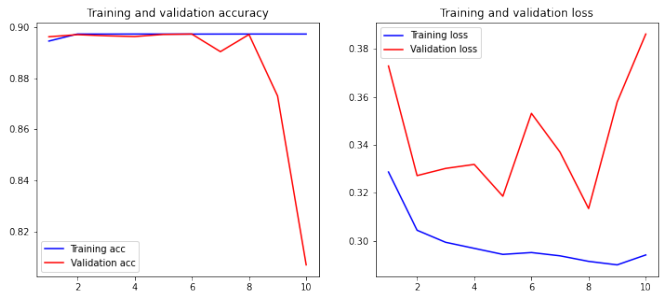

Fig. 3. Accuracy and Loss Curve of LSTM Model

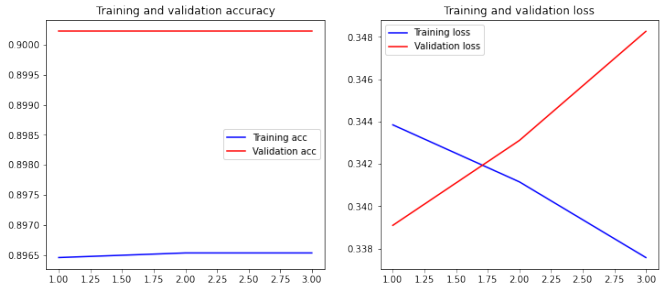

Fig. 4. Accuracy and Loss Curve of BERT Model

TABLE I
Raw Performance and Uncertainty-aware Performance of LSTM and BERT

Model	LSTM	BERT
Number of Parameters	194,818	109,724,579
Test Accuracy	89.75%	91.02%
Test F1-Score	80.69%	90.02%
Monte Carlo Ensemble Accuracy	87.0%	91.75%

In this section, we analyze the performance of our proposed models. We assess the models using a variety of performance criteria, such as test accuracy, test F1-score, and Monte Carlo ensemble accuracy. Table I shows the raw and uncertain-aware performances of LSTM and BERT.

The LSTM model was trained with weight parameters of 194, 818. During the evaluation, it attained an accuracy of 89.75% and an F1-score of 80.69%. Upon the application of Monte Carlo ensemble techniques, the model attains an ensemble accuracy of 87.0%. The BERT model, on the other hand, performed better despite having significantly larger weight parameters of 109, 724, 579. It outperformed the LSTM model in terms of accuracy and F1-score, attaining 91.02% and 90.02% respectively. Moreover, the model achieves an ensemble accuracy of 91.75%.

The training vs validation curves are shown in Figure 3 and in Figure 4. The difference in epochs is due to architectural requirements. BERT needs very few epochs to train. The underfitting issue occurs because of multiple dropouts. However, without dropout, the test performance drops, and overfitting becomes an issue.

Now, in comparison to their performances, it is apparent that both the Uncertainty-Aware LSTM and BERT models demonstrate improved robustness in fake review detection. BERT however exhibits superior performance. Nevertheless, it should be noted that BERT requires a significantly larger parameter count, implying its higher computational requirements for training and inference.

D. Uncertainty Analysis

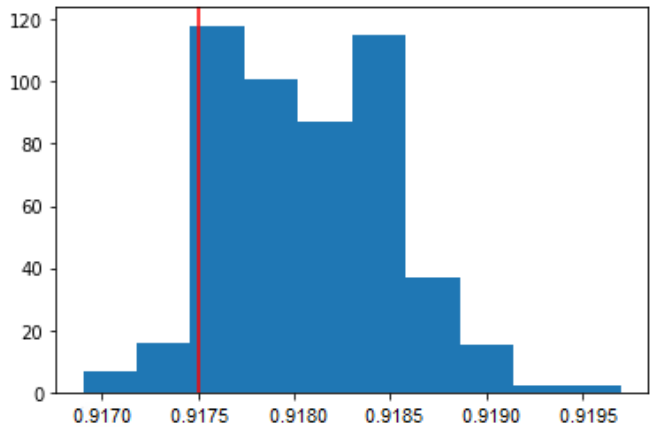

Fig. 5. Monte Carlo Accuracy (Blue) and Ensemble Accuracy (Red)

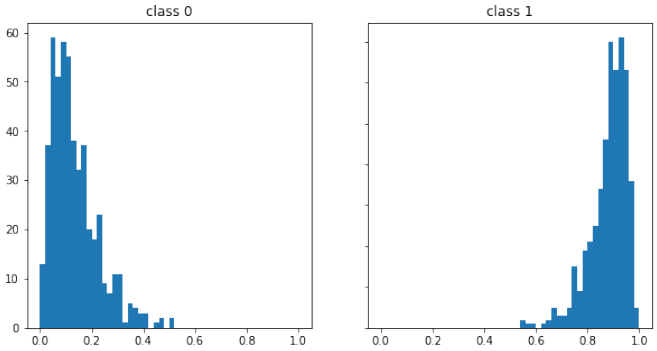

Fig. 6. A Certain Prediction Example of BERT Model

Uncertainty measurement tells us which model is more certain. In the previous segment, we show how confident BERT is, overall. Figure 5 shows the ensemble accuracy diagram of BERT. The addition of MCD makes the models more robust and reliable. For in-depth analysis, we follow the sample-by-sample analysis. Figure 6 and Figure 7 are the softmax-score distributions of two different samples. One is a certain prediction and the other one is an uncertain sample. Both of them predicted class '1' and the actual label is the same. Figure 6 is the certain prediction. It represents that the softmax-score of class '1' is more than 60% in the majority of cases. As a result, if most of these scores are in the right position of the predicted class then it is a certain prediction. Figure 7 is a different scenario, where both classes have similar distributions. Hence why this prediction is an uncertain one. For this sample, the mean softmax-score is

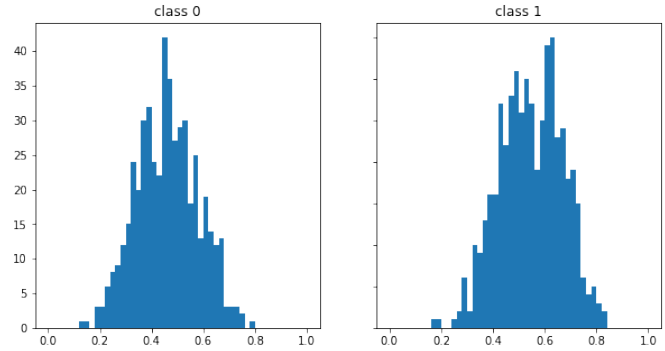

Fig. 7. An Uncertain Prediction Example of BERT Model

58.8%. On the other hand, the mean softmax-score is 88.1% for certain prediction cases. Here the model was tested 500 times with a 25% dropout rate with the dropouts being utilized randomly in different neurons or nodes of each layer. Thus, the robustness of the prediction is strengthened.

V. Conclusion & Future Work

Fake reviews have raised concerns among customers and online traders alike. Fortunately, advancements in machine learning have enabled us to detect them with the utmost accuracy. In our research, we utilize the evaluation metrics such as accuracy, F1-score, and binary cross-entropy loss to experiment with both LSTM and BERT models, as well as their Monte Carlo Dropout embedded variants. These models were evaluated using the Yelp Labelled Dataset, and the results obtained are rather promising. The BERT model surpasses the LSTM model in terms of accuracy and F1 score and achieves its highest accuracy when utilizing MCD. Our experimental model is novel because it is not only reliable but also incorporates uncertainty-awareness. In summary, our research contributes to the advancement of studies in the field of fake review detection by delving into the effectiveness of BERT and LSTM models when coupled with MCD techniques.

Looking ahead, our future plans involve incorporating the Switch transformer to further enhance the performance of our detection system. We aim to address more ambiguous cases, thereby providing more reliable predictions for real-world applications.

References

[1] R. Mihalcea and C. Strapparava, "The lie detector: Explorations in the automatic recognition of deceptive language," in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ser. ACLShort '09. USA: Association for Computational Linguistics, 2009, p. 309–312.

[2] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, ser. WSDM '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 219–230. [Online]. Available: https://doi.org/10.1145/1341531.1341560

[3] V. Sandulescu and M. Ester, "Detecting singleton review spammers using semantic similarity," in *Proceedings of the 24th International Conference on World Wide Web*, ser. WWW '15 Companion. New York, NY, USA: Association for Computing Machinery, 2015, p. 971–976. [Online]. Available: https://doi.org/10.1145/2740908.2742570

[4] A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "What yelp fake review filter might be doing?" in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 7, no. 1, 2021, pp. 409–418. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/14389

[5] M. Crawford, T. M. Khoshgoftaar, J. D. Prusa, and et al., "Survey of review spam detection using machine learning techniques," *Journal of Big Data*, vol. 2, p. 23, 2015.

[6] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," 2011.

[7] M. Mittal, I. Kaur, S. Chandra Pandey, A. Verma, and L. Mohan Goyal, "Opinion mining for the tweets in healthcare sector using fuzzy association rule," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 4, no. 16, p. e2, Oct. 2018. [Online]. Available: https://publications.eai.eu/index.php/phat/article/view/1280

[8] K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu, "Defend: Explainable fake news detection," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 395–405. [Online]. Available: https://doi.org/10.1145/3292500.3330935

[9] M. Albahar, "A hybrid model for fake news detection: Leveraging news content and user comments in fake news," *IET Information Security*, vol. 15, no. 2, pp. 169–177, 2021. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/ise2.12021

[10] D. Mehta, A. Dwivedi, A. Patra, and et al., "A transformer-based architecture for fake news classification," *Social Network Analysis and Mining*, vol. 11, p. 39, 2021.

[11] A. Q. Mir, F. Y. Khan, and M. A. Chishti, "Online fake review detection using supervised machine learning and bert model," *arXiv preprint arXiv:2301.03225*, 2023.

[12] R. Kaliyar, A. Goswami, and P. Narang, "Fakebert: Fake news detection in social media with a bert-based deep learning approach," *Multimedia Tools and Applications*, vol. 80, no. 8, pp. 11 765–11 788, 2021.

[13] C. I. Eke, A. A. Norman, and L. Shuib, "Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and bert model," *IEEE Access*, vol. 9, pp. 48 501–48 518, 2021.

[14] A. M. Elmogy, U. Tariq, A. Mohammed, and A. Ibrahim, "Fake reviews detection using supervised machine learning," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 1, 2021. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2021.0120169

[15] A. Shelmanov, E. Tsymbalov, D. Puzyrev, K. Fedyanin, A. Panchenko, and M. Panov, "How certain is your Transformer?" in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, Apr. 2021, pp. 1833–1840. [Online]. Available: https://aclanthology.org/2021.eacl-main.157

[16] M. F. Islam, F. Bin Rahman, S. Zabeen, M. A. Islam, M. Sabbir Hossain, M. H. Kabir Mehedi, M. Arafat Manab, and A. A. Rasel, "Rnn variants vs transformer variants: Uncertainty in text classification with monte carlo dropout," in *2022 25th International Conference on Computer and Information Technology (ICCIT)*, 2022, pp. 7–12.

[17] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ser. ICML'16. JMLR.org, 2016, p. 1050–1059.

[18] A. Meraj, "Yelp labelled dataset," https://www.kaggle.com/datasets/abidmeeraj/yelp-labelled-dataset, 2018.