Detecting AI-Generated Online Reviews in Amazon

Mark Kevin A. Ong Yiu
Department of Information Systems and Computer Science
Ateneo de Manila University
Quezon City, Philippines
markkevin.ongyiu@gmail.com

***Abstract*—The rapid growth of e-commerce has emphasized the significance of online reviews in consumer decision-making. However, the prevalence of fraudulent reviews poses significant challenges to the integrity of e-commerce platforms. In this study, we propose an approach that utilizes the Distilled-GPT-2 model to generate fake online reviews, then compares the performance of classical machine learning models (Random Forest and Support Vector Machine) and a deep learning model (BERT) in distinguishing between AI-generated and authentic reviews. Our findings suggest that while classical models initially perform well, they exhibit a decline in accuracy as the Distilled-GPT-2 model improves, whereas the BERT model demonstrates robustness across epochs. These results underscore the potential of transformer models in detecting fraudulent reviews and highlight the need for further research in this domain. Future studies could explore the adaptation of this approach across different product categories and leverage more advanced LLMs for data generation.**

***Index Terms*—AI-generated reviews, E-commerce, Fraud detection, LLM, GPT**

I. Introduction

A. Background

E-commerce (electronic commerce), whereby goods and services are traded over the internet, has grown significantly in the last few years. This increasing digitalization of the world economy has only been compounded by the influence of the COVID-19 pandemic [1], when many consumers were forced to transition online due to quarantine restrictions. The global revenue of the e-commerce market reached over $6 trillion in 2023, with estimates showing that one out of every four purchases will be done online by 2027 [2].

Given the size of this industry, there has been no shortage of research on factors that affect a consumer's decision to purchase a product or service. Several studies over the last decade have linked online reviews to various aspects of consumer behavior [3]–[9]. Online reviews have become an essential part of the consumer decision-making process. A survey in 2015 reveals that 82% of consumers read online reviews before deciding to purchase a product or service, with 93% reporting that online reviews affect their shopping choices [9].

However, this dependence on online reviews has repeatedly been exploited by bad actors, usually by creating fraudulent reviews to artificially increase or decrease ratings. In 2021, the world's leading e-commerce websites (including Amazon) have self-reported that, on average, 4% of all their online reviews are fake [10]. Translating this into economic impact, the direct influence of fraudulent online reviews on global e-commerce revenue in 2023 can be estimated to be $240 billion (or 4% of $6 trillion). These fraudulent reviews not only deceive consumers, but also distort market competition and misrepresent the true quality of products and services. Hence, it is imperative for e-commerce platforms to be able to detect and combat these fraudulent reviews.

The detection of fraudulent reviews traditionally rely on a mixture of manual inspection and heuristic-based methods. However, not only is this labor intensive, but it is also time consuming and difficult to scale. With the advent of large language models (LLMs) that are able to generate human-like text, with the most advanced models achieving super-human performance [11], bad actors are now able to create even more convincing fraudulent reviews, rendering the traditional approach for detection infeasible.

However, the same advancements in machine learning provide a growing opportunity to develop automated and more effective fraud detection methods. A common difficulty in the development of many fraud detection models is a lack of a proper dataset. Many studies still use heuristic-based methods to identify fraudulent reviews in the construction of their dataset [12], [13], and this potentially introduces various assumptions and biases on what constitutes a fraudulent review. However, this issue can be directly addressed by using LLMs to automatically generate fake reviews to construct a dataset.

B. Research Objectives

The objectives of this study are three-fold:

1) Fine-tune an LLM to generate fake online reviews.
2) Develop a model that is able to accurately classify whether or not a review was generated by the LLM, solely based on text content.
3) Investigate the effect of improving the fine-tuned LLM on the accuracy of our classification models.

This approach guarantees that a fake review in the dataset is truly fake, resulting in a more reliable detection model. Moreover, the dataset constructed in this study can also be used by future studies as a more reliable benchmark for models that aim to detect AI-generated reviews. Lastly, as LLMs continue to improve, it would be valuable to investigate its effect on our ability to distinguish its output from authentic reviews. Insights gleaned from this investigation can help

shape recommendations for strategies that combat review fraud in the age of LLMs.

C. Scope and Limitations

This study focuses on distinguishing AI-generated reviews from authentic reviews, regardless of whether or not the AI-generated review aligns with the true quality of the product that is being reviewed. Detecting other types of fraudulent reviews, such as reviews that misrepresent the true quality of the product or reviews that refer to the wrong product, are outside the scope of this study.

This study also focuses on the *Wireless* category of online Amazon reviews. However, the same framework can easily be adapted for other categories of reviews, and even other types of text-based data outside the domain of online reviews.

II. Materials

This study uses the *Amazon Customer Reviews* dataset, which contains over 100 million customer reviews of Amazon products from 1995 to 2015. This dataset was publicly hosted in Amazon's S3 bucket for some time, until they decided to withdraw support. However, copies of this dataset can still be found across the internet. In particular, this study uses the version of this dataset uploaded by Kaggle user *Cynthia Rempel* [14].

A. Overview

Due to its large size, the full dataset was partitioned into thirty-seven TSV (tab-separated value) files consisting of customer reviews from each pre-defined product category. Overall, all files combined contain over 50 Gigabytes (GB) of data and exactly $109,830,520$ instances of customer reviews across all categories.

Each line in the data files corresponds to an individual review, and all files have the same fifteen columns described in Table I. However, due to time and resource constraints, this study will focus mainly on the *Wireless* product category (one of the thirty-seven files). Since all the data files share the same columns, future studies can easily use the same workflows outlined in Section III on other product categories. The *Wireless* category was chosen in this study for no particular reason other than it was the default partition during the time when the dataset was hosted in Amazon's S3 bucket.

TABLE I
Columns in the Amazon Customer Reviews dataset

Column	Description	Data Type
`marketplace`	Two-letter country code of the marketplace where the review was written	string
`customer_id`	Random identifier that can be used to aggregate reviews written by a single author	string
`review_id`	The unique identifier of the review	string
`product_id`	The unique identifier of the product that the review pertains to	string
`product_parent`	Random identifier that can be used to aggregate reviews for the same product	string
`product_title`	Title of the product	string
`product_category`	Broad product category that can be used to group reviews	string
`star_rating`	A rating from 1 to 5 of the review	integer
`helpful_votes`	The number of helpful votes	integer
`total_votes`	The number of total votes	integer
`vine`	Review was written as part of the Vine program	categorical ("Y" or "N")
`verified_purchase`	Customer making the review was verified to purchase the product	categorical ("Y" or "N")
`review_headline`	The title of the review	string
`review_body`	The review text	string
`review_date`	The date the review was written	date

B. Wireless Product Reviews

The dataset contains exactly $9,002,021$ customer reviews on wireless products. It includes reviews for a variety of wireless products, from bluetooth headsets to car chargers.

An investigation into the distinct values for the `marketplace` column reveals that all instances have the same `marketplace` value of "US", indicating that all the reviews were written in the US marketplace. This may be relevant to the potential generalizability of our resulting models. Future studies may want to explore applying the same workflows to online reviews in another markerplace.

This dataset also has exactly $5,197,905$ distinct customers and $906,592$ distinct products. An interesting observation is that more than half of the unique customers in the dataset have only made one review. Similarly, more than half of the unique products in the dataset only have one review. This is important to consider when fine-tuning an LLM to generate fake online reviews, as it may not perform as well in these specific products/customers due to a lack of training data. Fig. 1 shows a box-plot of the number of reviews per customer. The box plot for the number of reviews per product is very similar, so it has been omitted from this paper.

Another important variable to consider is the star rating of the reviews. Fig. 2 shows the number of reviews in each star rating, which reveals that an overwhelming majority of the reviews in the dataset have a rating of 5 stars. To avoid any sentiment bias, we will apply some preprocessing to balance the number of reviews per star rating.

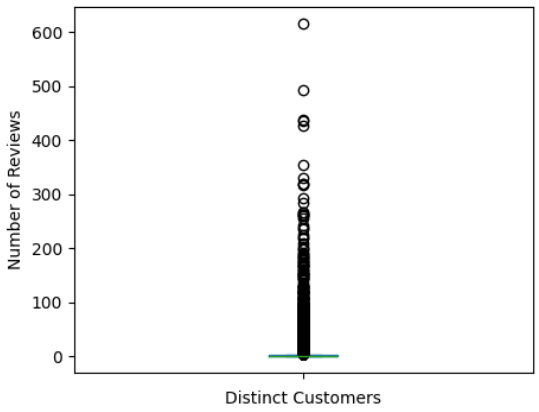

Fig. 1. Box plot of the number of reviews per customer

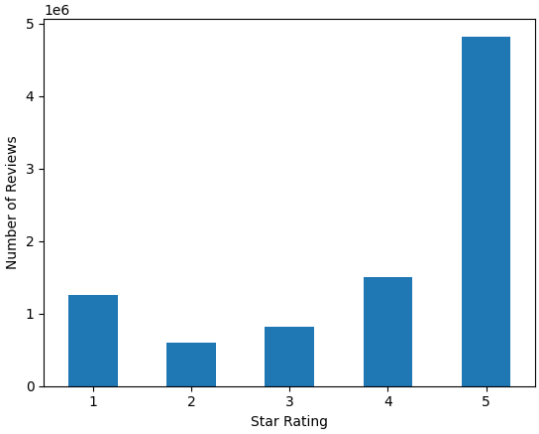

Fig. 2. Number of reviews in each star rating

III. Methods

A. Data Cleaning

Since we are generating the fake online reviews ourselves, we can guarantee that the reviews labelled as fake in our dataset is truly fake. However, we cannot simply assume that all the reviews in the *Amazon Customer Dataset* are authentic. This is because we expect about 4% of the reviews to be fake [2]. Since we do not yet have any classifier models at this phase, we can instead apply some generous heuristics to hopefully weed out the fake reviews, as well as make it easier for our LLM to train on the review data.

The first of these heuristics is the `verified_purchase` variable. If a customer has been verified to have bought the product that they are reviewing, then their review would more likely be authentic. Thus, we can simply remove all reviews that do not have a verified purchase. This is a viable heuristic because only a small portion of the reviews in the dataset are not verified. Fig. 3 shows the number of verified and nonverified reviews.

Another potential hueristic would be the `vine` variable. Vine is an invitation-only program which selects the most insightful reviewers in the Amazon store [15]. Thus, if a review

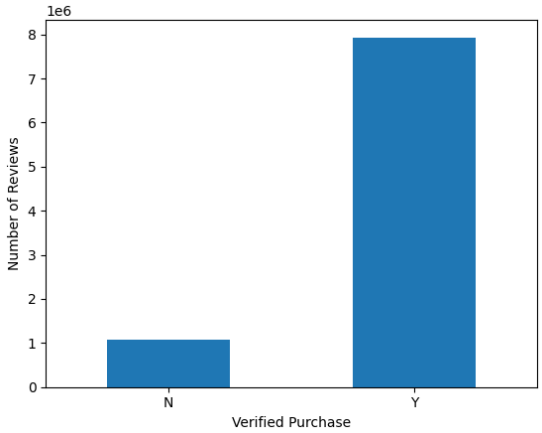

Fig. 3. Number of reviews with verified purchases

were written by someone in the vine program, then we can almost guarantee that the review is authentic. However, unlike `verified_purchases`, there is only a very small number of reviews written by customers in the vine program (see Fig. 4). If we were to remove all the nonvine reviews, then we would be left with a very small dataset. Thus, this hueristic is infeasible.

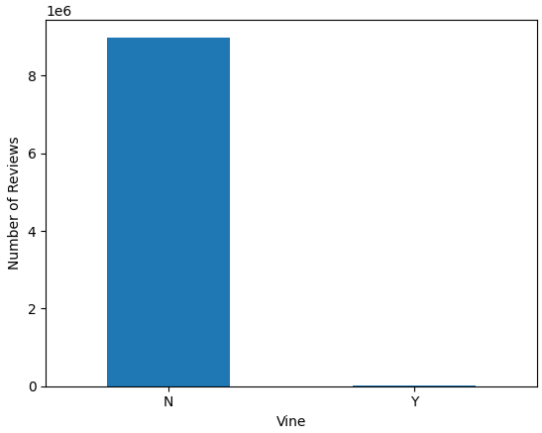

Fig. 4. Number of reviews under the Vine program

Finally, we look at the length (or number of characters) in the `review_headline` and `review_text` of each instance. This is shown in Fig. 5 and Fig. 6, respectively. Note, however, that both histograms are skewed to the right, indicating the existence of a small number of outliers with long headlines and texts. These long review may or may not be fraudulent, but removing them from the dataset is still beneficial. The reduction in the variability of the length of the reviews would result in faster training times as well as more consistent generation of fake reviews. Thus, we arbitrarility choose a cut-off length to prune the dataset. In this study, we choose to remove any reviews that exceed a headline length of 100 or a review length of 280. We also remove reviews that have headlines or texts that are exceedingly short. In this

study, we remove all reviews that have less than 10 characters in their headline or body.

To summarize, we prepare the dataset for data generation by first removing all reviews that do not have a verified purchase. Then, we remove all reviews that have a headline length exceeding 100 characters, as well as all reviews that have a review length exceeding 280 characters. Finally, we remove all reviews that have a headline length or review length less than 10 characters.

Finally, to avoid any bias in sentiment, we balance the number of reviews per star rating through undersampling. That is, we take the number of reviews in the star rating with the least reviews and sample that amount from all five star ratings.

Performing all these steps on the *Wireless* product category dataset results in a dataset with exactly $36,180$ reviews, and a perfectly balanced number of reviews per star rating.

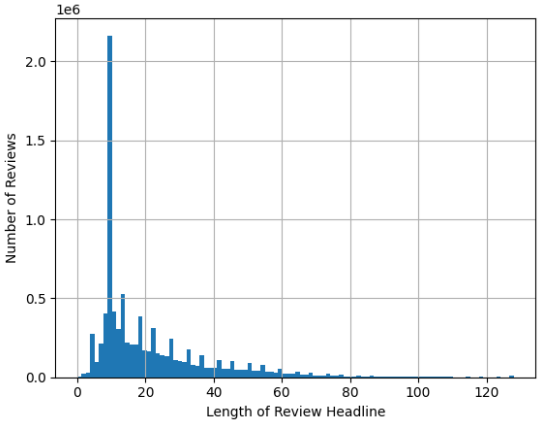

Fig. 5. Number of reviews per headline length

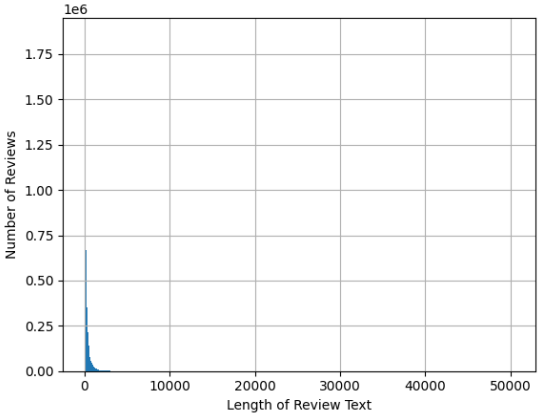

Fig. 6. Number of reviews per review length

B. Data Generation

To generate fake reviews, we fine-tune a Distilled-GPT-2 model on a stratified subset of the cleaned dataset (1,000 per star rating). Distilled-GPT-2 is an English-language model based on the Generative Pre-trained Transformer 2 (GPT-2) [16]. The distilled model was designed to be a faster and more lightweight version of GPT-2, with comparable performance due to knowledge distillation [17].

LLMs similar to GPT-2 work by predicting the next token in a sentence. It does this by taking a sequence of text and producing a probability distribution for the next token. It then produces the next token by sampling from that distribution. The new sequence is then fed back into the model to continue generating new tokens until the response is completed by an end-of-response token.

Thus, we can embed some control on the kind of review we want to generate by including a pre-defined prefix before every review. In this study, we want to be able to dictate the star rating of the review to be generated. Thus, we form the input sequence by using the template – "A `[star_rating]`-star review: `[review_body]`".

This will fine-tune the Distilled-GPT-2 model to generate responses in the form of that template. Then, to generate a fake review, we simply choose a star rating and ask the model to continue the sequence with the review body left blank – "A `[star_rating]`-star review:".

Using this approach, we can generate an arbitrary amount of fake reviews for each star rating. Reviews are generated until we have the same number of fake reviews as real reviews for each star rating. In this study, we generate 500 fake reviews per star rating and randomly sample 500 authentic reviews per star rating, resulting in dataset of $5,000$ reviews.

Note that we did not include the product title in the input template. This means that the product being reviewed will be randomly chosen by the LLM. This was an intentional decision as it makes our models less useful for bad actors. The product title may also be unreliable in this dataset as more than half of the unique products only have one review. Moreover, this study focuses on distinguishing AI-generated reviews from authentic reviews, so the product title was not necessary for the purposes of this study.

C. Models

In this paper, we will compare three classification models – two classical machine learning models and one deep learning model. In particular, we will compare a Random Forest (RF), a Support Vector Machine (SVM), and a transformer model.

A Random Forest (RF) is an ML algorithm that makes use of an ensemble of decision trees to make predictions. For classification tasks, individual decision trees predict the class labels of data points by following decision rules created based on the features of the data. Each tree in the ensemble votes for the class label and the RF model's final prediction is the class that receives the majority of the votes [18].

A Support Vector Machine (SVM) is an ML model that classifies datasets into two classes. In simple cases, such as in this study, it does this by fitting a hyperplane that best splits all the instances into the two classes [19]. However, more sophisticated boundaries may also be used depending on the kernel of the SVM.

Lastly, a transformer is a type of neural network that relies entirely on attention mechanism rather than recurrence [20]. Notably, transformers have been shown to perform really well on natural language processing (NLP) tasks. In fact, the best sentiment analysis model is a transformer, and many LLMs are based on transformers. In particular, this study uses Distilled-BERT, a distilled version of the Bidirectional Encoder Representations from Transformers (BERT) model developed by Google in 2019 [21].

D. Evaluation

In this study, three metrics will be considered for evaluating model performance, namely, accuracy, precision, and recall.

Accuracy measures the ability of the model to correctly classify instances. It is calculated by taking the ratio between the number of correct predictions and the total number of predictions, as described in

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{1}$$

Precision measures the accuracy of the model's positive predictions. It indicates the reliability of a "fradulent" prediction. It is calculated by taking the ratio between the number of correctly predicted "fraudulent" instances and the total number of "fraudulent" predictions.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

Recall measures the ability of the model to correctly classify an instance as "fraudulent." It indicates how well actually fraudulent reviews are correctly classified by the model. It is calculated by taking the ratio between the number of correctly predicted "fraudulent" instances and the total number of "fraudulent" instances.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

E. Training and Testing

After generating fake reviews using the fine-tuned Distilled-GPT-2, we split the dataset into two partitions, with 80% going to our training set, and the remaining 20% going to our testing set.

The training set is the data on which our classification models will be trained on, and hyper-parameters will be tuned using a 10-fold stratified cross-validation. We do not want to use the test set for hyper-parameter tuning as this may result in overfitting.

We also repeat the same workflow as we increase the number of epochs the Distilled-GPT-2 model is tuned for. This allows us to investigate the effect of improving LLMs on our classification models.

IV. Results and Discussion

The Distilled-GPT-2 model was tuned for 20 epochs and the workflow was repeated at every epoch. For the purposes of this discussion, we shall refer to the dataset produced by the Distilled-DPT-2 model after k epochs as the level-k dataset. For example, the dataset generated after one epoch would be referred to as a level-1 dataset, and the dataset generated after twenty epochs would be referred to as the level-20 dataset.

Fig. 7 shows the accuracy of the three classifiers on the testing set as the number of GPT epochs vary. The RF model was able to achieve an accuracy of 81.0% on the level-1 dataset, but quickly deteriorated to 72.0% on the level-20 dataset. The SVM model was able to achieve an accuracy of 80.8% on the level-1 dataset, but similarly deteriorated to 75.6% on the level-20 dataset. However, the BERT model was able to relatively maintain its accuracy across the different levels, with an accuracy of 84.3% at level-1 and 82.3% at level-20. While all models were still able to perform significantly better than random chance, a clear pattern of deterioration can be observed on the RF and SVM models.

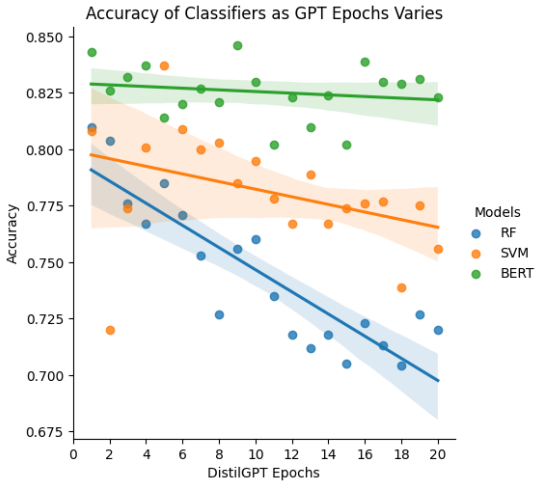

Fig. 7. Accuracy of Classifiers on the Testing Set

Fig. 8 shows the precision of the three classifiers on the testing set as the number of GPT epochs vary. We can observe the same general pattern. The RF model has a precision of 93.8% at level-1, but quickly deteriorates to a precision of 67.7% at level-20. Similarly, the SVM model has achieves a precision of 84.4% at level-1, but only a precision of 72.4% at level-20. On the other hand, the BERT model achieves a precision of 79.5% at level-1 and a precision of 75.5% at level-20. While both classical models outperform the BERT model at level-1 in terms of precision, the BERT model quickly catches up by level-20 as it retains its performance while the classical models deteriorate.

Fig. 9 shows the recall of the three classifiers on the testing set as the number of GPT epochs vary. Here, we observe that the recall appears to increase as the level increases. However, the BERT model significantly outperforms the classical models across all levels. The RF model achieves a recall of 66.4% at level-1 and a recall of 84.0% at level-20. Similarly, the SVM model achieves a recall of 75.6% at level-1 and a recall of 82.8% at level-20. Impressively, the BERT model was able to achieve a recall of 92.4% at level-1 and recall of 95.6% at level-20.

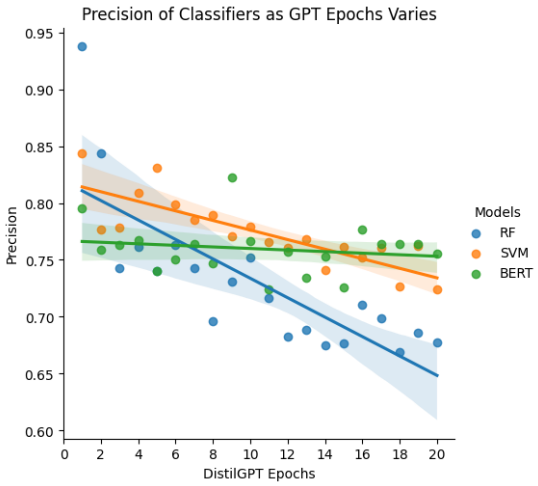

Fig. 8. Precision of Classifiers on the Testing Set

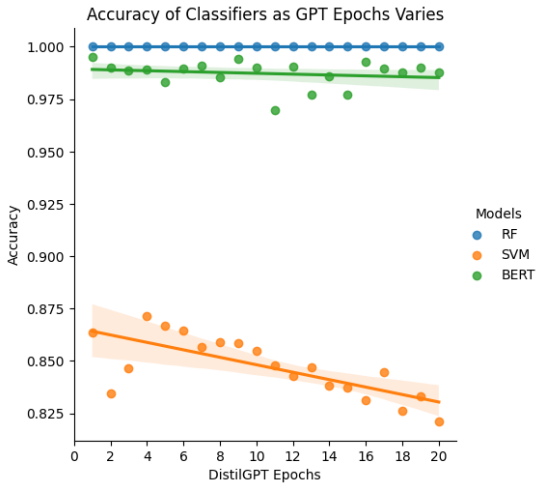

Fig. 10. Accuracy of Classifiers on the Training Set

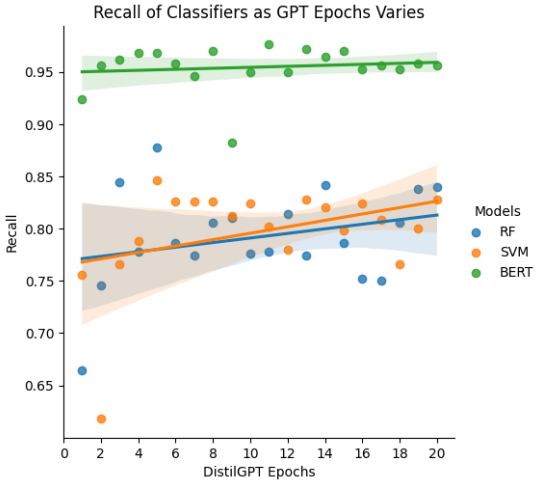

Fig. 9. Recall of Classifiers on the Testing Set

Lastly, Fig. 10 shows the accuracy of the three classifiers on the train set as the number of GPT epochs vary. Comparing this with Fig. 7 can tell us whether or not our models over-fitted on the training set. Here, we can observe that the RF model clearly over-fitted as it achieved perfect accuracy across all levels. Similarly, the BERT model also over-fitted (although not a bad as the RF model) as it achieved a much higher accuracy on the training set compared to the accuracy on the testing set. On the other hand, the SVM model appears to not have over-fitted as it achieves a similar accuracy on the training and testing sets. This makes sense as L2 regularization was applied to the SVM as part of its hyperparameters.

V. Conclusion

Overall, our classification models we able to accurately distinguish between AI-generated reviews and authentic reviews. While the accuracy deteriorated for the RF and SVM models, we still achieved an accuracy of around 70% or greater on almost all levels. However, the BERT model still performed the best among the three classification models.

It is also important to note the robustness of the BERT model compared to the two classical models. While the two classical models (RF and SVM) deteriorate in performance as the dataset level increases, the BERT model maintained its performance even as the Distilled-GPT-2 improves.

This result can have profound consequences to the development of review fraud detection models. It tells us that perhaps classical models are no longer viable to detect fraudulent reviews in this age of LLMs. However, larger and more rigorous studies must be done before we can make definitive conclusions. This study uses an relatively outdated LLM to produce its fake reviews. It would be interesting to see how state-of-the-art LLMs would fare in this study.

VI. Recommendations

This study has made many sacrifices due to time and resource constrains. As such, there are many opportunities for improvement and further research.

First, the Distilled-GPT-2 model was only trained on a subset of $5,000$ reviews for 20 epochs. It would be interesting to see the results if all $36,180$ reviews were to be used for more epochs. In particular, it may be possible that the BERT model would begin to deteriorate if we were to train the Distilled-GPT-2 model for more epochs.

Second, this study only focused on the *Wireless* category of Amazon reviews. It would be interesting to perform the same workflow on other categories and test our models across different categories. For example, it would be interesting to see if our classification models, that are trained on the *Wireless* category, would fare well on a dataset produced by the *Apparel* category. It would also be interesting to see how our workflow can be adapted to other text-based domains outside of online reviews.

Third, this study did not perform hyperparameter tuning on the BERT model, which resulted in some overfitting (see

Fig. 10). Future studies may attempt to apply some regularization techniques to the BERT model to see whether or not performance still be improved.

REFERENCES

[1] S. Khumalo, M. M. Mlotshwa, Z. K. Khumalo, and O. R. Raphalo, "The Impact of COVID-19 on e-Commerce Through a Systematic Review," *European Conference on Innovation and Entrepreneurship*, vol. 18, no. 1, pp. 462–468, Sep. 2023. [Online]. Available: https://papers.academic-conferences.org/index.php/ecie/article/view/1439

[2] "2023 & 2024 eCommerce Stats, Trends & Forecasts ⁄ Artios," Nov. 2023. [Online]. Available: https://artios.io/ecommerce-statistics/

[3] K. Z. Zhang, C. M. Cheung, and M. K. Lee, "Examining the moderating effect of inconsistent reviews and its gender differences on consumers' online shopping decision," *International Journal of Information Management*, vol. 34, no. 2, pp. 89–98, Apr. 2014. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0268401213001588

[4] C. Ruiz-Mafe, K. Chatzipanagiotou, and R. Curras-Perez, "The role of emotions and conflicting online reviews on consumers' purchase intentions," *Journal of Business Research*, vol. 89, pp. 336–344, Aug. 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0148296318300274

[5] B. Von Helversen, K. Abramczuk, W. Kopeć, and R. Nielek, "Influence of consumer reviews on online purchasing decisions in older and younger adults," *Decision Support Systems*, vol. 113, pp. 1–10, Sep. 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167923618300861

[6] J. Guo, X. Wang, and Y. Wu, "Positive emotion bias: Role of emotional content from online customer reviews in purchase decisions," *Journal of Retailing and Consumer Services*, vol. 52, p. 101891, Jan. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0969698918309160

[7] T.-C. Kang, S.-Y. Hung, and A. H. Huang, "The Adoption of Online Product Information: Cognitive and Affective Evaluations," *Journal of Internet Commerce*, vol. 19, no. 4, pp. 373–403, Oct. 2020. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/15332861.2020.1816315

[8] Y. Wu, T. Liu, L. Teng, H. Zhang, and C. Xie, "The impact of online review variance of new products on consumer adoption intentions," *Journal of Business Research*, vol. 136, pp. 209–218, Nov. 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S014829632100494X

[9] T. Chen, P. Samaranayake, X. Cen, M. Qi, and Y.-C. Lan, "The Impact of Online Reviews on Consumers' Purchasing Decisions: Evidence From an Eye-Tracking Study," *Frontiers in Psychology*, vol. 13, p. 865702, Jun. 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2022.865702/full

[10] "Fake online reviews cost $152 billion a year. Here's how e-commerce sites can stop them," Aug. 2021. [Online]. Available: https://www.weforum.org/agenda/2021/08/fake-online-reviews-are-a-152-billion-problem-heres-how-to-silence-them/

[11] OpenAI, "GPT-4 Technical Report," 2023. [Online]. Available: https://arxiv.org/abs/2303.08774

[12] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding Deceptive Opinion Spam by Any Stretch of the Imagination," 2011. [Online]. Available: https://arxiv.org/abs/1107.4557

[13] V. Sandulescu and M. Ester, "Detecting Singleton Review Spammers Using Semantic Similarity," 2016. [Online]. Available: https://arxiv.org/abs/1609.02727

[14] "Amazon US Customer Reviews Dataset." [Online]. Available: https://www.kaggle.com/datasets/cynthiarempel/amazon-us-customer-reviews-dataset

[15] "Amazon Vine." [Online]. Available: https://www.amazon.com/vine/about

[16] "distilbert/distilgpt2 · Hugging Face," Apr. 2023. [Online]. Available: https://huggingface.co/distilbert/distilgpt2

[17] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," 2019. [Online]. Available: https://arxiv.org/abs/1910.01108

[18] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001. [Online]. Available: https://doi.org/10.1023/A:1010933404324

[19] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, ser. COLT '92. New York, NY, USA: Association for Computing Machinery, Jul. 1992, p. 144–152. [Online]. Available: https://dl.acm.org/doi/10.1145/130385.130401

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," 2017. [Online]. Available: https://arxiv.org/abs/1706.03762

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018. [Online]. Available: https://arxiv.org/abs/1810.04805