Reviewbox

Reading Reviews like a Pro

An Algorithmic Review Fraud Detection

Project Lead and Manager: Khaled

Customer Success Manager: Tejasvini

Technical Lead Analyst: Iris

Technical Mentor Coordinator: Yvette

Business mentor Coordinator: Jordon

Meetings Coordinator: Daviid

Event Planner: Nora

Table of Contents

Abstract

Online shoppers rely heavily on product reviews to make their purchasing decisions. However, there is a growing problem of fraudulent reviews that mislead shoppers through false portrayal of a product's quality. Thus, detecting fraudulent reviews has been in the spotlight of e-commerce operations. This report covers a methodology to detect fraudulent reviews on Amazon. After identifying key features associated with fraudulent behavior, a machine learning classifier model was built to detect fraudulent reviews which have been deleted from Amazon with a 92% accuracy.

Background

The Review Problem

On e-commerce platforms, customers use reviews as a reference when making purchasing decisions. Products with higher volume of sales and higher rating of reviews will earn higher ranks, and more exposure to the buyers. However, this competition draws bad players. Certain vendors might resort to hiring agencies which specialize in generating fake reviews on platforms like Amazon [1]. Those fraudulent reviews can give either good or bad ratings, aiming to improve one's product and harm competitors' products. As those behaviors harm the authenticity of a product, Amazon has designed an algorithm to detect fraudulent reviews. Once Amazon has identified a fake review, they would take it back and assign low weights to the single review's score which contributes to the product overall rating [2].

Although Amazon has algorithms to detect such fraudulent behavior, their identification is not timely for every product on its platform. Moreover, their algorithm is likely not comprehensive and may not capture all the fraudulent reviews. For these two reasons, the Reviewbox team aims to utilize available Amazon data to employ a predictive model that will identify the fake reviews.

A Focus on Incentivized Reviews

Since Oct 2016, Amazon has deleted over 500,000 reviews, 71% of which were incentivized. These incentivized reviews are usually a result of vendors offering gifts in exchange for positive reviews. The average rating for these deleted reviews was 4.75 stars. Most incentivized reviews, around 95%, are not verified by the purchaser. While Amazon has effectively began acting on incentivized reviews, there are still reported incentivized reviews that appear on listings. Reviewmeta, an available online tool that detects fraudulent reviews, reports approximately 1.5% incentivized reviews from over 10 million analyzed reviews [3].

Problem Introduction to Reviewbox

The severity of this problem was introduced to Reviewbox by one of their customers whose brand was significantly impacted due to unfair competition [4]. To properly structure this problem, fraudulent reviews were categorized into two separate classes:

1. Untrusted or suspicious reviews: These reviews are determined by identifying key features that may be associated with suspicious reviews such as repetitive phrases or unverified purchases.

While one feature on its own may not be plausible to claim a review as untrusted, a review that is flagged for multiple features might less likely be a result of natural review behavior.

2. <u>Incentivized reviews:</u> By definition, incentivized review is when a vendor on Amazon sends product to customers as a gift in return for good reviews. Identifying such behavior on its own is plausible enough to report as it is a clear violation of Amazon's selling policies [5].

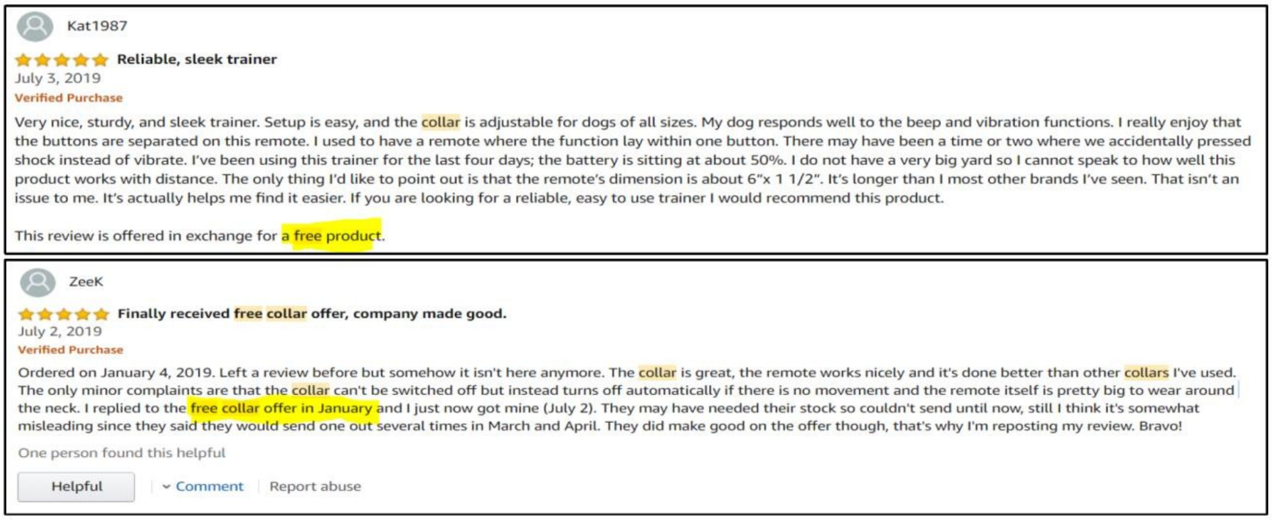

Figure 1: Incentivized Review

Approaching the Problem

Available Solutions

In building a predictive algorithm, a model input and a model output is required. The output for this problem is clear: identifying whether a review is fraudulent or not. The input on the other hand is not so straightforward. Thus, the assessment of the input variables to classify a fraudulent review constituted the foundation of the entire analysis. Understanding the characteristics associated with fraudulent behavior is crucial and looking at available solutions is a helpful starting point.

The Fakespot Model

One of the most popular solutions is an online tool provided by Fakespot [6]. Through proprietary technology, Fakespot evaluates the authenticity of reviews of a product listing by assessing:

- Language pattern recognition
- Reviewer profile
- Correlation with other reviewer information

An algorithm is trained to pick up patterns associated with suspicious behavior to generate an output with a score from 'A' to 'F' for an entire product listing. 'A' denotes a product with trusted reviews and an 'F' denotes a product with suspected fraudulent reviews.

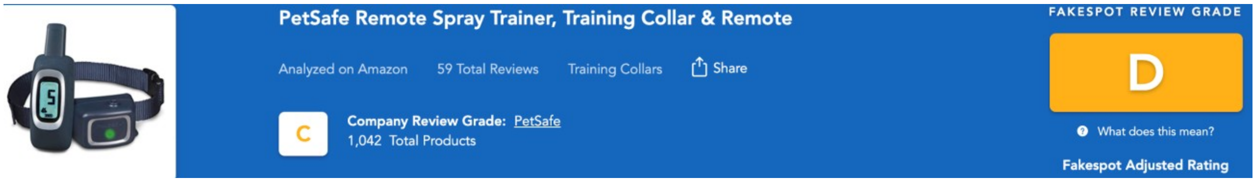

Figure 2: Fakespot Product Score

The Reviewmeta Model

One of Fakespot's competitors is Reviewmeta [7]. They provide an adjusted product rating (similar to Amazon's) after omitting suspicious reviews. This helps consumers get a sense of what a product rating should "actually" look like.

Figure 3: Reviewmeta Product Score

They also provide individual review scores for a select number of reviews as following:

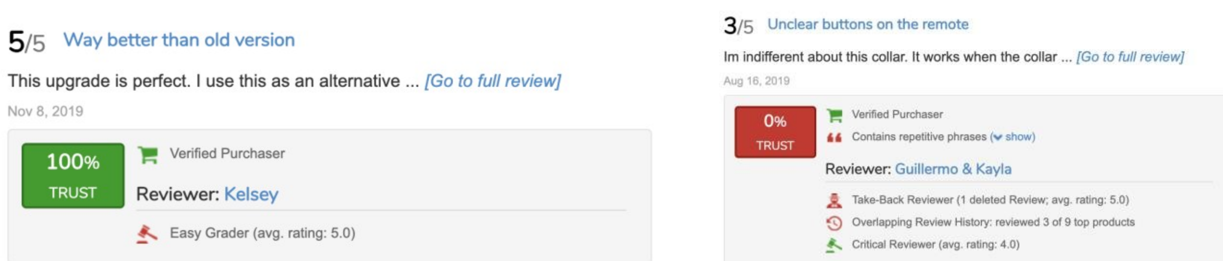

Figure 4: Reviewmeta Individual Review Score

Individual review ratings "Trusted" or "Untrusted" are provided while also displaying the features contributing to these ratings. For example, in the untrusted review on the right, one of the features flagged was "Take-Back Reviewer" which means that the reviewer has a history of having their reviews deleted. This is one of several features that contributed to the given score rating.

Key Features

To identify fraudulent behavior on Amazon, it is best to begin by understanding the motive behind people posting inauthentic reviews. There are several reasons such as incentivization to receive free products or to boost a brand's image on the online marketplace. The real challenge is that reviews can often look authentic when viewed alone but could change when clustering them with a pool of similar reviews. To elaborate on this point, we look at the two examples below:

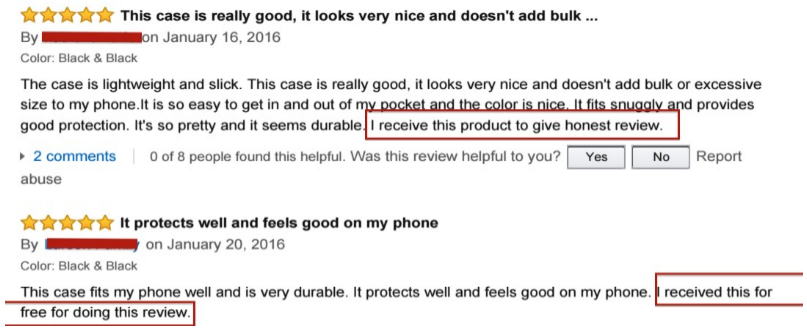

Figure 5: Example #1 [8]

Ruling: Fraudulent

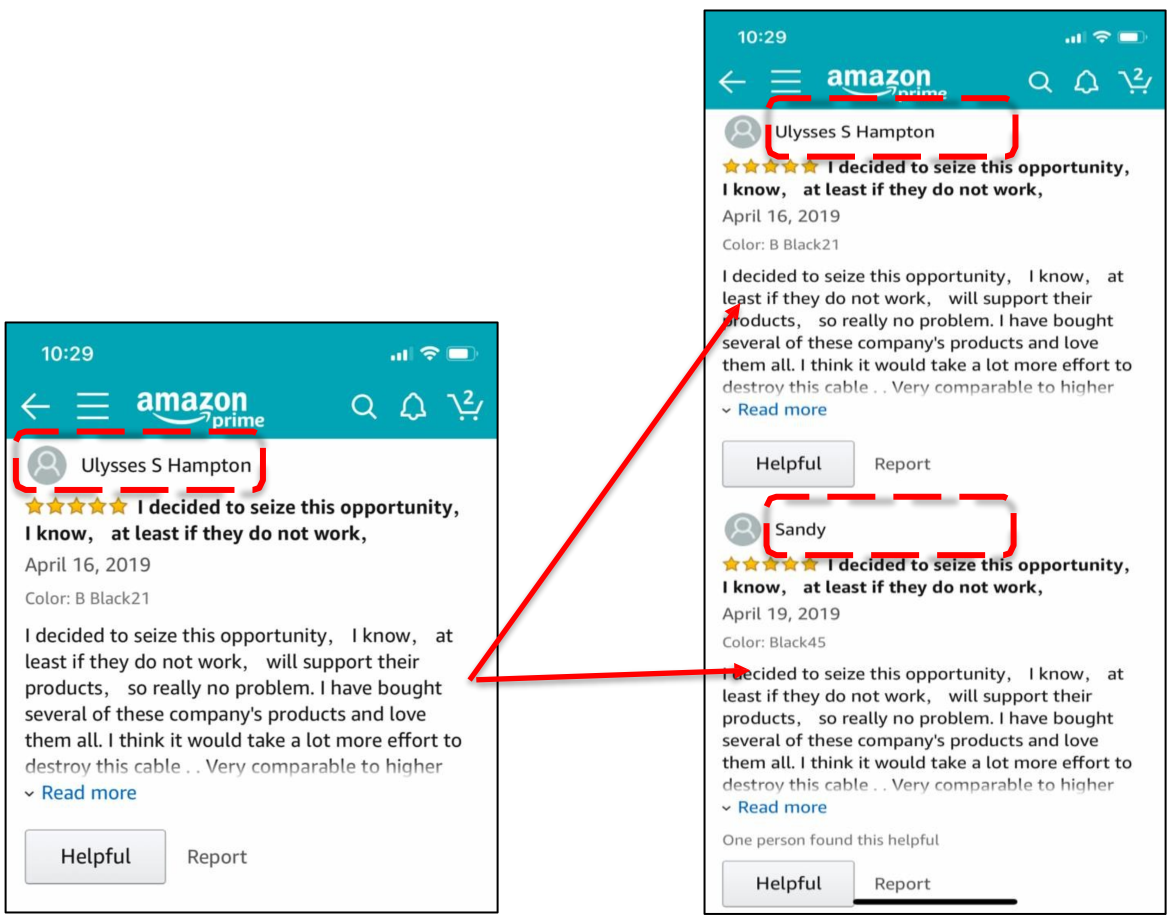

Figure 6: Example #2 [9]

Ruling: ~~Authentic~~ Fraudulent

Based on the examples above, it is noticed that classifying reviews on an individual level is not always a solution. To properly assess whether individual reviews are fraudulent, they must be examined across multiple features with a pool of reviews. But which features are important when assessing reviews?

Feature engineering was a critical part in this project. A well-rounded assessment of reviews is required by considering any aspect that may be associated with fraud. Through extensive domain research, 16 key features were generated, of which 13 were used for the final model to classify the review authenticity. These features are categorized as profile-related and non-profile-related features.

Profile Related Features

1. *Single review:* Indicates that the reviewer has only posted a single review on Amazon signaling inactive user behavior.
2. *Take back review:* Indicates that a user has a history of deleted reviews.
3. *Never-verified reviewer:* Indicates that the reviewer has never made verified purchases on Amazon.
4. *Single day review:* Indicates a reviewer posted all their reviews on the same date.
5. *Easy reviewer:* This feature is created based on two conditions: First, the average rating of the reviewer is greater than or equal to 4.5. Second, the reviewer gave a 5-star review for this purchase.
6. *Same date:* This feature indicates whether a reviewer has posted 20 reviews or more on the same date.
7. *Overlapping review history:* Research shows that companies are sometimes hired to mass-review specific products to boost a brand's image. The overlapping review history features tackles this by identifying reviewers with overlaps in reviews.
8. *0 review:* Indicates that the reviewer's profile has no reviews displayed.
9. *Brand loyalist (3 features):* This feature is divided into three with similar concepts. The insight is whether the profiler's reviews are primarily associated with the subject product brand. This was done by creating three features with different thresholds:
    a. Minimum: Check if profiler has more than 1 review for the same brand
    b. Medium: Check if 50% or more of the profiler's reviews are for the same brand.
    c. Maximum: Check if 100% of the profiler's reviews are for the same brand.

    Due to statistical insignificance, the three brand loyalist features were excluded from the final model.

Non-profile Related Features

1. *Non-verified purchase:* Indicates whether the reviewer's purchase was a verified purchase on Amazon. Non-verified purchases are more likely to be inauthentic.
2. *High volume day:* This feature is produced by looking at the history of the product reviews received over time. By setting appropriate thresholds, days at which there is an influx of reviews can be determined. Reviews generated on these days are then flagged as high-volume day reviews.
3. *Overrepresented word count:* This feature categorizes review content into bins (for e.g. 11-15 words or 16-25 words per review). It then compares the distribution of word counts to the

distribution of review word counts of the product **category**. If the word bins distributions are mismatching by a set threshold, reviews within the associated word bins are then flagged as having overrepresented word count.

4. *Incentivized review:* This feature is generated by identifying reviews which contain terms associated with incentivization such as “free gift in exchange for a review”.
5. *Repetitive phrases:* This was one of the most complex features produced using Natural Language Processing techniques. The aim of this feature is to flag reviews that have similarity in phrases like Figure 6 above.

Methodology

By understanding the business problem, the data required to perform the analysis was identified. Reviews should be mapped to the key features of interest. After which, this dataset may be fed into an appropriate machine learning model that would produce an output stating whether a review was trusted or fraudulent.

To be able to train a model to identify fraudulent reviews, Reviewmeta’s existing concept was reproduced. By collecting enough data on Reviewmeta’s rated reviews, the key features associated with fraud (returning a 0% trust score) were recognized. Based on the determined feature importance, the model could then be inverted back to Reviewbox’s customer datasets to provide similar ratings. The steps are outlined below:

1. Collect Reviewmeta individual review scores / associated features
2. Prepare Reviewmeta data in a suitable format to feed into our desired model
3. Train a predictive model based on a considerable amount of data points
4. Test on unseen data to measure model accuracy
5. Extract Reviewbox’s customer dataset and map features similar to step 2 format
6. Invert tested model from step 4 onto Reviewbox’s customer dataset

Data Preparation

This section explains the details of **steps 2 and 5** highlighted above. The desired predictive model will consist of several independent variables and a single dependent variable. The independent variables are the features highlighted in the previous section and may be considered as flags that will return a value of 1 when picked up. For example, the incentivized review feature will be flagged as 1 if the review contains terms associated with incentivization.

Step 2: Reviewmeta Data

To explain this step in detail, we may look at the review below:

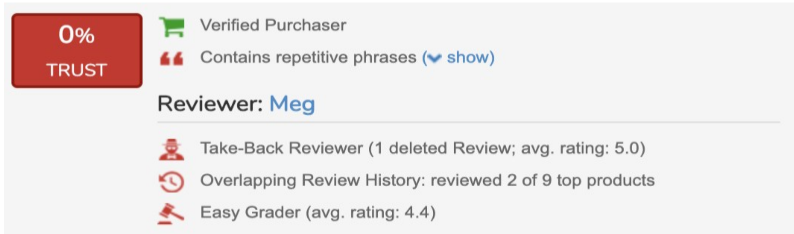

Figure 7: A Reviewmeta Individual Score

This dataset, prepared on an individual review-level, for the predictive model will have:

1. A unique review id or text
2. 12 independent variable columns (for all features)
    a. Repetitive phrases will be flagged as 1
    b. Take-back reviewer will be flagged as 1
    c. Overlapping review history will be flagged as 1
    d. Easy grader will be flagged as 1
    e. All other features will be set to 0
3. 1 dependent variable column with value of 0 signifying no trust

This step is repeated for over 10,000 reviews across different product categories to obtain adequate training data with a satisfactory model accuracy.

Step 5: Reviewbox Data

Unlike the Reviewmeta data, the Reviewbox customers data has not yet been mapped to the features of interest. The data behind those features come from disparate sources:

1. Profile-related features
    a. Amazon's reviewer profile page
2. Non-profile related features
    a. Reviewbox Reviews Dataset
    b. Reviewbox Sales Rank Dataset

Profile-related features were obtained by collecting reviewer profiles pages based on the profile ids provided on Reviewbox's Reviews Dataset. Later, the features of interest were extracted from the collected pages. For an individual reviewer / profile ID, the profile features (9 independent variables) are collected.

Non-profile related features are pertinent to the review content rather than the profiler content and thus are mainly obtained from the Reviews Dataset. The Sales Rank Dataset is also used to merge product categories for relevant features relevant features such as overrepresented word count. Once those features are obtained, they are merged with the profile-related features on a common column such as the review text or profile id.

Similar to the Reviewmeta data, the final dataset will consist of a unique review id or text column, 13 independent variable columns and 1 dependent variable column.

The First Approach

Initially, the aim was to evaluate Reviewmeta's existing solution to see if it meets the customers' requirements. By collecting enough data points for a large sample of rated reviews, the weights that Reviewmeta assigned to individual features was better understood.

One of the primary challenges was that products are more likely to have a large amount of trustworthy reviews (imbalanced data). Thus, it would be hard to obtain enough data points that contained the flagged features of interest. Nonetheless, this was mitigated for using the *Amazon Bad Actors (Competitive Reviews) Report* which contains a list of products with suspicious activities. In addition to researching suspicious products online, a dataset of 21,000 reviews was collected, out of which approximately 5,000 were untrusted (0-10% score).

Once the data was prepared in the appropriate format, it was time to begin developing the algorithm. The model was designed to train approximately 70% of the data and test or predict the remaining unseen data points. Several model approaches were evaluated:

- XGBoost model
- Ordinal logistic regression model
- Generalized linear model with logit link function

Based on the findings, the XGBoost Model was able to produce the best results by predicting the testing data with an accuracy of 78%. Essentially, this meant that the XGBoost model was able to assign appropriate weights to the 12 independent variable features to predict the correct classification of a review with a 78% accuracy.

The Second Approach

The model in the first approach was able to reproduce Reviewmeta's classification of reviews with a 78% accuracy. To further improve the model accuracy, the number of training data was increased by almost double. Additionally, more features were added to the model to better represent the data. Nonetheless, the accuracy did not increase.

To investigate this problem further, individual reviews were reviewed thoroughly to understand how the classifications were generated. It was revealed that while some reviews were very similar with respect to flagged features, they were rated extremely differently.

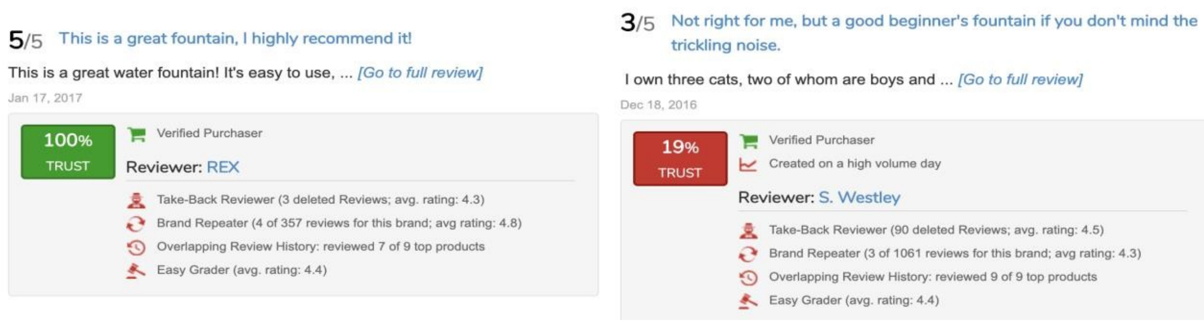

Figure 8: A Reviewmeta Individual Score (Trusted vs. Untrusted)

Looking at the above reviews, it is difficult to justify the difference in rating with similarly flagged features. Since the features (independent variables) were still a valid part of the analysis, new ways to improve labelling of reviews (dependent variable) were explored.

Looking into individual reviews collected from Reviewmeta, it was noticed that some of the reviews that were collected months ago were deleted.

"Sorry, we couldn't find the page."

There was a possibility that such reviews were deleted by Amazon due to suspicious behavior. In that case, review labels may be improved by assigning only deleted reviews as fraudulent reviews. This does however raise a concern as a review may be removed in one of two situations:

- Fraudulent review deleted by Amazon
- Review deleted by the reviewer themselves

To increase the possibility of the first case, only reviews that were deleted and given a trust score lower than 10% by ReviewMeta were considered. By adding this constraint, a strong hypothesis may be formed that such reviews were deleted due to fraud.

Detecting "Fraudulent" Amazon Deleted Reviews

With this hypothesis in mind, the dataset for the classification model was prepared to detect the fraudulent reviews deleted by Amazon. There are 3 steps involved this process:

1. <u>Get deleted reviews, good reviews and combine them together:</u>

   First, the Reviewmeta dataset collected is filtered using a threshold of trust score smaller than 0.1. Then, reviews which are deleted on Amazon are identified. By subtracting the full dataset and the deleted reviews dataset, a set of "good" reviews that were not deleted is obtained.

   Accordingly, approximately 13,500 good reviews and 900 bad reviews are obtained. Simply combining them together for analysis would be a problem due to an unbalanced dataset. Thus, a sampling is performed on the good review dataset, specifically, a stratified sampling. This helps build a stronger classifier because if only good reviews with a 100% trust score are included, the reviews might not have many flags. That would make it "easier" for the classifier to pick out fraudulent reviews. After the sampling was done, approximately 2,000 good reviews and 900 bad or deleted reviews are obtained.

2. Collect the profile pages to create more features:

   Initially, the independent variables used in the model were primarily based on the ones created by Reviewmeta. To build a stronger classifier, profile pages were collected to create more profile-related features. By introducing more features, the model was better explained to enhance accuracy in predicting fraudulent reviews.

3. Remove reviews with only 1/2 flags to improve model performance:

   During the initial attempts to train the model, a very low precision score was generated, meaning the model is falsely picking out many authentic reviews and classifying them as fraudulent. To tackle this problem, the reviews that were deleted but only have 1 or 2 flags were omitted. This made the model stricter and greatly improved the precision score. After this adjustment, the final dataset deduced contained approximately 2,000 good reviews and 700 bad or deleted reviews.

The Classifier

After the above three steps are completed, a dataset for model training is prepared. This is split into training and testing groups. The training dataset will train the model while the test dataset will be used to evaluate the out-of-sample performance of the model. Random Forest and XGBoost models were used because of their high performance on classification problems. For each model, Gridsearch was applied to optimize the hyperparameters that would ultimately optimize the model performance.

Hyperparameters: In machine learning, a hyperparameter is a parameter whose value is set before the learning process begins. The hyperparameters are tuned before the model training process starts to enhance the model performance.

Gridsearch: It is the process of performing hyper parameter tuning to determine the optimal values for a given model. Normally, multiple possible values for each hyperparameter is provided and it will exhaustively try every possible combination and choose the best set of values.

Thus, in total four models are created:

1. Random Forest (RF) with default hyperparameters (HPs)
2. Random Forest (RF) with an optimized set of hyperparameters (HPs)
3. XGBoost with default hyper parameters (HPs)
4. XGBoost with an optimized set of hyperparameters (HPs)

Model Evaluation & Selection

After training the model, the performance was evaluated using several key metrics: AUC score, confusion matrix, recall rate and precision rate.

1. AUC Score: Explains how much the model is capable of distinguishing between classes. The higher the AUC, the better the model is at predicting zeros as zeros and ones as ones [10].
2. Confusion Matrix: A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class [11].
3. Recall Rate: Attempts to answer the question: What proportion of actual positives was identified correctly? In this case, it asks of all the fraudulent reviews, what proportion of them are we able to identify?
4. Precision Rate: Attempts to answer the question: What proportion of positive identifications were "actually" correct? In this case, it asks of all the reviews that have been classified as fraudulent, what proportion of them are "actually" fraudulent?

Among these metrics, AUC score and recall rate are the most important. The classifier should be able to distinguish fraudulent reviews and authentic reviews as much as possible. Meanwhile, it should also pick out as many fraudulent reviews as possible.

Here are the metrics on the test dataset for the four different models:

	RF with default HPs	RF with optimized HPs	XGBoost with default HPs	XGBoost with optimized HPs
AUC Score	91%	92%	90%	92%
Recall Rate	88%	92%	86%	89%

Table 1: Model Evaluations

From table 1, we can see that the Random Forest model with optimized hyperparameters performs the best as it is getting both the highest AUC score and the highest recall rate. The precision score of it is just slightly lower than that of the other two models. Accordingly, this model was selected for deployment.

Incentivized Reviews as a Product

Why build a separate incentivized reviews product?

Initially, incentivized review was just one feature that contributed to the Reviewscore model. As the project progressed, this feature showed clear importance as it was self-explainable to report to Amazon for further action. An indicator of reviews on Reviewbox platform could help customers quickly identify incentivized reviews posted on their competitors'' products.

Target

While Amazon has been actively removing incentivized reviews, there are still many new reviews that are incentivized that can affect fair competition in the marketplace. It has been reported that there are still 0.5%-1.5% reviews that are incentivized [2]. The target is to label those reviews and send notifications to Amazon for deletion.

How it works

Once Reviewbox has automatically collected data 3 times a day, the incentivized algorithm is run to detect incentivized patterns. Once detected, a team will label those features and send notifications.

Algorithm explained:

Stage 1: Retrieve the initial list of incentivized reviews

A word list is crafted from observed incentivized reviews and researches. After using text data treatment, text analysis is performed to find the existence of incentivized terms. When a match is found, an initial label is assigned.

Stage 2: Improve its accuracy

After deploying the first step, 0.5% of the reviews are usually labeled as incentivized. However, after manually checking individual reviews, only 50% of them are found to be incentivized.

An automated method which does not resort to manually labelling all reviews retrieved from stage 1 is required. For that, features were crafted and used in a predictive model to remove false positive results filtered out by the first stage.

By building a random forest classifier on engineered features, the final product was able to extract incentivized reviews with a 90% accuracy.

Limitations

The model was built based on research and practical assumptions. The strength of the analysis was based on features determined using domain knowledge. Nonetheless, there are some possible biases in the approach:

1. **Sampling Bias**

   The training data was collected based on the assumption that if a review was deleted by Amazon, it is fraudulent. To increase the probability of only capturing fraudulent reviews, reviews with lower trust scores were selected; however, there is no way to ensure the deleted reviews selected were 100% deleted by Amazon.

   More observations are needed. Currently, the dataset used to train the review score model includes approximately 2,700 datapoints with 700 deleted reviews. One way to improve the model reliability in the future is including more data points, especially deleted reviews.

2. **Feature Engineering Bias**

   Features were extracted using collected profile pages. However, the current method might not seamlessly capture some features. For some features such as take-back reviews, the number of reviews in a certain profile page was counted and compared with the number displayed in the reviewer's profile. The reviewer was flagged when a difference in numbers was spotted. A more reliable way is to monitor the review history of reviewers over time. Whenever a review is deleted, it can be properly recorded.

Another way to help increase the prediction power of the model is to create more meaningful features. For example, Reviewmeta uses the take back reviews rating as a feature; however, it was not included in the model due to limitations in the available data. To create it, the reviewer profile pages would need to be collected in different time periods. The reviewer's history is then compared at different time points to get the take back reviews and average take back review rating.

3. **Modeling Bias**

   It is difficult to measure the effectiveness of the created model as we are unable to observe if the reviews flagged are then deleted by Amazon. To further support this analysis, it would be best to track flagged reviews, which are reported, over time to verify if they are eventually deleted by Amazon.

Model Deployment

Reviewscore

In the final model deployment, the following datasets are required:

1. Reviewbox's review dataset including an additional column with the profile id.
2. Profile pages data from the profile ids in the above dataset.

Once the above datasets are provided, the features used for the classifier are then created to structure a final dataset identical to that of the created model. Accordingly, the feature importance may be applied into Reviewbox's reviews dataset to tag individual reviews (fraudulent or non-fraudulent). The final model will not only help output review classification but will also highlight the features flagged to contribute to this rating.

Incentivized Reviews as a Product

Future products may be examined for incentivization by applying the classifier model on Reviewbox's review dataset. Upon uploading a dataset with ASIN's of interest, a 2-stage process is automatically executed to shortlist reviews with incentivization features.

Once the above models are deployed, individual reviews with their associated labels (fraudulent/incentivized) may then be extracted and integrated within platform's review register.

Conclusion

While Reviewbox currently utilizes large sums of data to help brands monitor their products' performance through descriptive analytics, the aim of this product is to introduce advanced analytics to the existing tools. Business on e-commerce platforms is more important today than before with the increased reliance on on-demand services and increased restrictions on visits to retail stores due to the COVID-19 outbreak.

With such increased reliance, it is vital to ensure that Reviewbox's brands are having fair market play to avoid jeopardizing lost revenues. By building a product to tackle fraudulent reviews, Reviewbox is able to expand their platform services to their customers for stronger insights. More importantly, this will reduce product hampering by competitors who are using techniques violating trading policies. This will also set a pedestal for further advanced analytics to be utilized on Reviewbox's platform in the future through enhanced predictive and prescriptive analytics.

References

[1] Lomas, Natasha. “Facebook Still Full of Groups Trading Fake Reviews, Says Consumer Group.” TechCrunch, TechCrunch, 6 Aug. 2019, techcrunch.com/2019/08/06/facebook-still-full-of-groups-trading-fake-reviews-says-consumer-group/.

[2] Johnson, Tara. “The End of Amazon Incentivized Reviews - What's Next?” *Tinuiti*, 23 Sept. 2019, tinuiti.com/blog/amazon/end-amazon-incentivized-reviews-whats-next/.

[3] Perez, Sarah. “Amazon Makes Good on Its Promise to Delete ‘Incentivized’ Reviews.” *TechCrunch*, TechCrunch, 23 Nov. 2016, techcrunch.com/2016/11/23/amazon-made-good-on-its-promise-to-delete-incentivized-reviews/.

[4] Weissman, Cale Guthrie. “'We Are Getting Our Ass Kicked': Amazon Sellers Grapple with a Fake Review Problem.” *Modern Retail*, 8 Apr. 2020, www.modernretail.co/platforms/we-are-getting-our-ass-kicked-amazon-sellers-grapple-with-a-fake-review-problem/.

[5] “GP.” *Amazon*, Goettsche Partners, 2011, sellercentral.amazon.com/gp/help/external/G1801?language=en_US.

[6] “Analyze and Identify Fake Reviews and Counterfeits.” *Fakespot*, www.fakespot.com/.

[7] “ReviewMeta Analyzes Amazon Product Reviews and Filters out Reviews That Our Algorithm Detects May Be Unnatural.” *Review Meta: Amazon Reviews You Can Trust*, reviewmeta.com/.

[8] “New Amazon Ban on Incentivized Reviews.” *Annex Cloud Blog*, www.annexcloud.com/blog/new-amazon-ban-on-incentivized-reviews/.

[9] “The Fake Reviews Problem & How to Remove False Reviews.” *Chatmeter*, 23 Apr. 2019, www.chatmeter.com/blog/the-problem-with-fake-reviews-and-how-to-stop-them/.

[10] Narkhede, Sarang. “Understanding AUC - ROC Curve.” *Medium*, Towards Data Science, 26 May 2019, towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5.

[11] “Confusion Matrix in Machine Learning.” *GeeksforGeeks*, 23 Feb. 2020, www.geeksforgeeks.org/confusion-matrix-machine-learning/.