THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF INFORMATION SCIENCES AND TECHNOLOGY

EXAMINING CASES OF PROPERTY MISINFORMATION IN AIRBNB

VAISHALI DEVARAKONDA
SPRING 2022

A thesis
submitted in partial fulfillment
of the requirements
for a baccalaureate degree
in Data Sciences
with honors in Data Sciences

Reviewed and approved* by the following:

J. Andrew Petersen
Associate Professor of Marketing
Thesis Supervisor

John Yen
Professor-in-Charge of Data Sciences
Honors Adviser

Mallory M. Meehan
Assistant Clinical Professor of Real Estate
Faculty Reader

* Electronic approvals are on file.

ABSTRACT

Sharing economies are defined by their peer-to-peer approach to conducting essential business activities and are often facilitated through a platform. These platforms are also well known to depend on consumer product reviews to drive their operations. Airbnb, an online platform for lodgings and rentals, follows this model. Those who put up their spaces online are also able to rent others' properties for a given amount of time. Most reviews for Airbnb listings are positive, but there can be a discrepancy between customer ratings and listing descriptions that results in guests alleging property misinformation. While this phenomenon is rare, it can have significant downstream consequences for Airbnb and hosts who market on the platform. Thus, I look to create a model that determines when cases of property misinformation occur on the platform and assess the implications of this phenomenon for Airbnb. Overall, classes of misinformation are most accurately predicted with an ensemble of BERT-based classifiers, and the downstream consequences of misinformation are significantly negative in terms of revenue generation for Airbnb and hosts with property misinformation in their descriptions.

TABLE OF CONTENTS

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank Dr. John Yen for his continued guidance on my thesis endeavors and in navigating my academic life at the Schreyer Honors College. His advice has helped me appreciate the vast applications of Data Science in business and other fields. In addition, I would like to thank Dr. J. Andrew Petersen for his support in our weekly meetings which helped me translate my vision into a tangible solution. To Professor Mallory M. Meehan, thank you for your interest in my thesis and helping me gain expertise in jargon used in the Business world. Finally, I would like to thank my friends and family for their continued support during the thesis writing process and college career. They have all helped me keep a level head even when I felt overwhelmed through these past few semesters.

Chapter 1

Introduction

Airbnb is a platform that allows hosts to rent out their homes for potential guests. Users can also use the website to look for available rentals based on their needed location, size, amenities, and availability (Lau, 2015). However, there is a possibility that rentals can be misrepresented by inconsistencies between subjective listing descriptions and the reality that guests experience. When examining this discrepancy, the first word a reader would think of is "fraud". However, that is not what this paper is looking at when examining misinformation across the company's platform, as fraud is an illegal activity that Airbnb already combats within its platform. Rather, this paper looks to identify cases when there are differences in how the host describes the property and the guests' perceptions of the property based on the experience during a stay.

1. Motivation for Examining Property Misinformation

While Airbnb does have fields for hosts to input values like the number of bedrooms or the number of people that can comfortably use the space, most guests that use the platform see the description first. Highlighting descriptions can have two possible effects. They allow for more bookings, as guests expect a stellar experience from reading the host's description of the property. However, using attention-grabbing language, and then under-delivering on the promised stay, could cause guests to feel as if they were misled with a more subjective description of the property and experience that the listing offered.

The following examples from the data I used in this work illustrate just how common and nuanced the issue of misinformation can be. In Austin, Texas, a listing of a “Lux Modern House” was described as an “awesome luxury duplex” with the space being “almost completely renovated”. However, reviewers noted that “only half the house was renovated, but on the listing, it very much leads you to believe that the entire house is renovated”. In another case, a San Francisco property had the following description: “This nice bedroom with private bathroom and separate entrance is located on the first floor of a two-story home in San Francisco's Portola District”. However, a few guests noted that they were not notified about the space being located “in the basement/garage on his post until [they] booked, [which] felt a bit misleading.”

In theory, these examples suggest that property misinformation is an important issue that Airbnb needs to address. But given how rare property misinformation really is on Airbnb’s platform, examining the implications of misinformation could be more complex in practice. One reason for this is the platform’s 360 review system. In this system, guests can leave reviews about the host, property, or experience. Additionally, the host can leave a review on the guest. The process goes as follows: after a guest checks out of their stay, hosts use the platform to leave a review about a guest (*Airbnb*, n.d). This review cannot be edited and is shared after *both* the guest and host send in their reviews through Airbnb (*Airbnb*, n.d.). As a result, future guests or hosts will be able to determine whether the other party generally gets positive or negative reviews for previous stays. Since both the guest and host are impacted by Airbnb’s review policy, guests might not be encouraged to post reviews alleging property misinformation for the fear of being marked as a bad guest on Airbnb’s platform, when the host’s listing might in fact be misleading.

Another reason contributing to the rarity of property misinformation on Airbnb is that Airbnb already has guidelines on how hosts should list their property, and this includes directions on reflecting accurate listing information on the platform. However, these guidelines are vague, with one example being to have an "accurate and up-to-date address", which is typically shared with the guest *after* booking, allowing for some possible misinformation (*Airbnb*, n.d). Another expectation for an accurate listing is that the "photos fairly represent the condition and layout of the space" (*Airbnb*, n.d.). However, out of the reviews I found alleging property misinformation, many note that this was not the case. Findings like this will be further explored in this thesis.

These reasons show that property misinformation is still an issue that Airbnb needs to address for the platform to continue effective operations. Thus, addressing misinformation would allow for more clarity for hosts on what constitutes an "accurate" listing. This understanding will hopefully have a waterfall effect on guests and Airbnb itself, with guests enjoying their stay even more and Airbnb growing their user base of new and repeat guests over time. For the hosts on the platform, they will be able to refer to clearer guidelines that help include more objectivity in their description while not sacrificing the writing style that makes a listing attractive to begin with.

Given these motivations, I want to answer the following questions specific to Airbnb in this thesis:

- How often does factual misinformation i.e., property misinformation occur on Airbnb?
- How can one identify factual misinformation i.e., property misinformation in Airbnb's platform?
- What kinds of Airbnb properties are more susceptible to property misinformation?

- What are the downstream consequences of misinformation on a platform like Airbnb?

2. Approach to Addressing Research Questions

These questions were answered with various statistical analyses methods as well as more robust Machine Learning techniques. The first two research questions specified in this section relate to Natural Language Processing techniques, specifically through detecting property misinformation through the experiences of an Airbnb guest. To answer them, I utilized various Text Classification Machine Learning pipelines to determine which models would be best suited to detect property misinformation from the guest reviews of a given property. In Question 3, I focused on determining what features of the Airbnb data I obtained were most statistically significant to identify properties as having misinformation. As an example, one such feature could include a host's cancellation policy, as it could be hypothesized that properties without one would be more inclined to exhibit the phenomenon of property misinformation. In Question 4, I analyzed the revenue differences between properties with and without misinformation to determine the negative impact of misinformation on Airbnb and its hosts.

The rest of this paper will be organized as follows. I discuss related literature to this topic in the next section, focusing on other studies done on Airbnb review data as well as studies utilizing Natural Language Processing techniques. The Empirical Application section focuses on the technical work I have done on this topic, where I explore supervised classification techniques to answer the research questions above. I will then discuss the implications that my empirical results pose for Airbnb's platform, speaking to any changes that need to be made or considerations that the platform would need to consider addressing misinformation. Finally, this

paper is more of a starting point for addressing property misinformation in Airbnb, so I will conclude this thesis with a discussion of future avenues for research on this topic as well as some limitations in my methodology.

Chapter 2

Background

1. Related Works – Airbnb and Machine Learning Techniques

When it comes to text classification, a vast amount of literature exists in this space. Text Classification and Analytics algorithms have been developed and refined for the better part of the 21st century, and the creation of sharing economy platforms such as Airbnb has made these methods evolve in a myriad of different ways. This section will focus on some seminal works on Text-based Classification that motivated my study, with a special focus on using text documents such as customer reviews to answer classification questions.

In terms of Text Classification techniques, Netzer, Lemaire, and Herzenstein (2017) apply these methods to a completely different area from marketing. More specifically, the authors focus on leveraging loan request texts from borrowers to predict whether a given borrower is more likely to default on their loan when funded. Netzer et al. (2017) used a dataset from Prosper, a global crowdfunding online platform, and focused on loan requests that were funded between April 2007 and October 2008. More specifically, they worked with text variables from the request description, such as the number of characters or the percentage of words with at least six characters, in conjunction with the financial and demographic data of the anonymized borrower.

Netzer et al. (2017) completed their machine learning experiments using a similar approach to mine, as will be discussed in later sections of this paper. More specifically, they trained various classification models such as Logistic Regression and Random Forest Trees and used the weights of all models to determine the best models to include in an ensemble method

that works with all three categories of data. This technical approach worked well, as the team's ensemble model resulted in the best predictive power, measured by an AUC-ROC curve.

One advantage that Netzer et al.'s (2017) methodology may have is related to the three classes of data. While textual data alone had a very high level of predictive ability, augmenting the classifiers by adding financial and demographic data increased the accuracy of the model and its predictions. However, they note that one limitation is that the ensemble provided "little to no interpretation of the…words and topics that predict default" (Netzer et al., 2017). My thesis will address this limitation by examining more state-of-the-art methods that are geared to take the context of words and sentences into account.

Another important work using text classification methods comes from Luss and d'Aspremont's (2015) work in predicting *intraday* returns from financial assets through news articles from PRNewswire and historical return data from the NYSE Trade & Quote database at Wharton's School of Business. This work addressed the limitation of market volatility systems often ignoring financial news and looks to predict whether asset returns were expected to go up or down *and* whether the returns were greater than an expected threshold after a given time of the news release. Luss and d'Aspremont (2015) focused on looking at Support Vector Machine classification, a technique I experimented with, to determine whether returns are directed in a positive or negative direction. While they were able to classify the magnitude of returns easily using Support Vector Classification methods, the text preprocessing method is different from my work.

Rather than using tokenization techniques, which make text easier for the model to process, the authors used a "Bag-of-words" approach and created vector representations of the documents which housed the counts for each word. This method, while easy to implement, poses

a risk in that the context is removed from each word during the processing stage. I address this limitation in my work through using tokenizing methods geared for the models I choose to work with, as explained in Section 3C.

While this thesis focuses on using customer reviews to examine property misinformation, customer review data can be used for other purposes and situations as well. A notable example would include Ghose and Ipeirotis (2006), who use customer reviews to see how review subjectivity influenced the sale of products on Amazon's wholesale platform. The team used a mix of product and sales data from Amazon as well as customer reviews for each product in the dataset. They employed a subjectivity estimation technique that allowed them to determine the average probability of subjectivity. They also determined the deviation probability, or how likely a review was to deviate from a "subjective" class, as some reviews could contain both objective and subjective content. Using these two probabilities in conjunction with information like the number of product reviews, Amazon's product price, and the average rating, the team calculated the log value of the sales rank, which referred to a predicted increase or decrease in sales.

Overall, Ghose and Ipeirotis (2006) found that an increase in review subjectivity led to an increase in sales *depending on the product* and that reviews with a mix of objective and subjective content increased sales due to the perception that these were more informative. This research on predicting sales from customer reviews is like mine in that it relies on review subjectivity and language constructs to determine the occurrence of a given objective (i.e., an increase in sales). However, Ghose and Ipeirotis (2006) is mainly oriented to help product manufacturers improve their product or marketing processes, while this thesis focuses on the impact on a company *and* its users.

Another work that indirectly has an impact on customers is the study that Ma, Zhang, Yan, and Kim (2013) completed to determine how to leverage customer reviews for topic modeling methods. Essentially, they used a Latent Dirichlet Allocation (LDA) model leveraging a synonym lexicon to determine the specific product features a customer's review talked about. Their dataset focused on Chinese-language reviews, as the sentence and language constructions of Chinese are different from those of English. A notable finding here was that that the combination of their LDA model and synonym lexicon proved to be more accurate and precise than Association Rule Mining, another preferred model for topic modeling. While I do not use the methods discussed in this paper for my thesis, it is still an important work to consider in the realm of customer review data. In addition, using an LDA opens the door to the possibility of topic modeling for Airbnb. This can be explored in a future study.

2. Technology Overview

This section will investigate technologies I used in my work. Given the nature of the first two questions I am looking to examine, many of the techniques I used are typically employed in classification problems. In these situations, models predict whether a given instance (e.g., weather patterns, reviews of a given Airbnb listing) belongs to a certain class (e.g., sunny or rainy; alleged misinformation or not). There are two categories of models I chose to explore. The first category consists of more commonly used models, which can be thought of as general-purpose classifiers suited for any task. This section also investigates more State-of-the-Art techniques which are specifically designed for classification tasks

A. Common Machine Learning Models for Text Classification

Classification models typically aim to assign an instance of data to one of two, or one of multiple, classes. There are typically three types of classification methods that one could use depending on the task he or she is trying to accomplish: binary, multiclass, and multilabel.

Binary classification models are the simplest of the three, looking to classify data into two possible groups: 1 (affirmative) or 0 (negative). For example, if one looked at a customer review for an Airbnb listing, they could classify it as 1 (alleging property misinformation) or 0 (not alleging property misinformation). Multiclass classification techniques use a similar logic, but instead of having only two options, they have X classes to choose from. Multilabel classification is a bit more complicated in that an instance of data can belong to multiple classes out of a set of X choices. For the purposes of my methodology, I focused strictly on Binary classification models.

The first model I chose was a Naïve Bayes classifier. In essence, this model can be used for both binary (two-class) classification tasks and multi-class classification tasks by employing Bayes Theorem, which states the likelihood of event A occurring (i.e., a review classified as alleging misinformation) given the occurrence of event B (i.e., the review containing one of the aforementioned keywords). However, the "Naive" nature of this model comes from the fact that all features are assumed to be independent of each other. The model is very simple to build and implement, but it is regarded among the Machine Learning community as a "bad estimator" (Ray, 2017). Thus, I simply used this model as a baseline to estimate the scores around which the next few models will fall or pass.

Another Binary classification model one could investigate is a Logistic Regression algorithm. Unlike the term "regression" suggests, this model is a classification algorithm that

works for tasks with a binary target variable, multiclass variable, or an ordinal variable (e.g., film ratings from a scale of 1 to 5) (Raj, 2020). According to an IBM article on this classification method, “A logistic approach fits best when the task that the machine is learning is based on two values, or a binary classification” (IBM, n.d.). This was the perfect introductory model to experiment on with my data, as I specified two possible classifications for a guest review. The model assumes that all the data is linearly separable. In the case of this thesis, there exists a linear boundary above or below which certain sentences or words of reviews, when expressed as numerical tokens, fall. The position of each tokenized review indicates whether the review falls into the “alleges misinformation” or the “not alleging misinformation” category.

Looking into more robust techniques, one could employ a Random Forest classifier. This model uses multiple decision trees, each with a different set of features, to classify a given object into a class. The reason for using multiple trees with a different set of features is to find out what features in the data provided are most related to the final classification (Yiu, 2021). In the context of my data, once each review is converted into a set of tokens (i.e., a numerical representation of each word), different sets of tokens could be used as features to feed into the decision trees to make a prediction about whether a given review alleges misinformation or not. The final classification of a review is based on the most common prediction among all decision trees in the model. One thing to note is that the Random Forest model assumes that all trees are uncorrelated with respect to the features used, which would not be the case for sentences, as the numerical token value of a certain word depends on the values of those around it (Yiu, 2021).

Continuing with the decision tree logic, the Extreme Gradient Boosting (XGBoost) model uses a different implementation of the concept. XGBoost uses Gradient Boosting, a common model optimization function in Machine Learning, in a manner like parallel computing across

multiple machines, resulting in more efficient time usage (Brownlee, 2016). However, one key assumption this model makes is that the input data is tabular, which is not often the case when looking to classify text documents.

Finally, one of the most robust classification models is achieved through utilizing Support Vector Machines. More specifically, I refer to Linear Support Vector Classification (LinearSVC). This model works by finding the most significant input data features without using the logic of decision trees, as Random Forest does. Given any features that are difficult to classify, this model works to find the best linear decision boundary between classes by reducing the number of features (e.g., tokenized word features) until a clear separation of classes is reached. One assumption this model makes is that if the classifier can distinguish between the data points that are hardest to classify (e.g., reviews that contain language with a neutral sentiment), then it can work much better to classify the data that is easier to examine (i.e., reviews that clearly indicate anger or disappointment). Text data is often linearly separable, as there exist various tools to represent text as numerical tokens and has many document and word features to work with (Kowalczyk, 2014). Thus, I felt dimensionality reduction could be helpful for this research.

Table A shows a summary of the methods I used. This table refers to the pros and cons of each model and provides examples of how to use each model.

Model	Advantages	Disadvantages	Examples
Naive Bayes	• Easy implementation • Does not need much training data • Highly scalable	• Assumes feature independence • Zero-frequency problem	Simple implementation of Naive Bayes to determine whether to play tennis
Logistic Regression	• Efficient training • Easy implementation	• Assumes linearity between label and features	Logistic Regression with the Framingham Heart Study

	• No assumptions of class distribution • Quick to classify unknown records		
Random Forest	• Reduced overfitting for decision trees • Flexibility between classification and regression • Rule-based approach	• More computational power needed • Longer training time	Credit Card Fraud Detection (Section 13)
XGBoost	• High flexibility • Parallel processing techniques • Cross-validation techniques over multiple iterations of model	• Does not work on sparse/unstructured data • Gradient Boosting very sensitive to outliers in data	House Price Prediction using XGBoost
LinearSVC	• Works well with clear margin of separation between classes • More effective in high-dimensional spaces • More memory-efficient	• Not suitable for larger datasets • No probabilistic explanation of classification • If number of features > number of samples, model underperforms	Question Classification Based on Cognitive Levels using Linear SVC

Table A: Summary of Common Machine Learning Methods Used in this Thesis

B. State-of-the-Art Text Classification Models

While the above approaches can be useful to solving the issue of detecting property misinformation, their main disadvantage comes from the fact that they are more useful as general-purpose models. In essence, this means that while these models can solve a large variety of classification problems, they cannot handle certain types of data, such as text data or image data, without extensive preprocessing. This preprocessing is done in the form of converting words to vector representations, or tokens, but each word is considered an independent entity, separate from all other words surrounding it. However, language construction rules often note

that a word's meaning is based on the context of the sentence in which it is used. This is where more State-of-the-Art techniques come in.

For Natural Language Processing, one useful model category is the family of Bidirectional Encoder Representations from Transformers, or BERT for short. Originally outlined in a paper published by Google AI Language in late 2018, these models do work with processing text features for deep learning purposes, but they consider the context of each word simultaneously (Vajpayee, 2020). For example, the word "disappointing" in an Airbnb guest review can be assigned a weight of X in one review, but the same word can be weighted differently in another review depending on the context of the sentence the word is located in.

One thing that makes these State-of-the-Art models attractive is that BERT's architecture is simple, consisting of a set of encoders which work to transform the input sentence to usable features for Natural Language Processing Tasks. In addition, models using the BERT architecture are pre-trained on a set of existing data, which removes the need for the programmer to define the vector transformation process.

A helpful resource to explore more State-of-the-art models was the website HuggingFace, which is a large online repository containing information on State-of-the-art models suited for a large array of objectives, such as text classification. The website allowed me to filter through models based on the task I wanted to accomplish (i.e., text classification) as well as some other parameters, including the language of the data, the dataset a model was trained on, etc. After looking through this website, I found two models that were useful to answer the question of identifying property misinformation.

The first model to keep in mind is CardiffNLP's twitter-roBERTa-base-sentiment model, found on HuggingFace's model repository. This model is of the best possible choices for this

thesis due to the size of the pretrained data and the fact that the data's language matched my own (English). What really makes this model a state-of-the-art technique is that one can use it to accomplish 7 different tasks: emoji detection, emotion analysis, hate detection, irony detection, offense detection, sentiment analysis, and stance detection. One assumption that I made with the model selection here was that tweets have more significant sentiment scores. In other words, I conjectured that tweets, much like most reviews, are often worded strongly, so this model's ability to assign an accurate sentiment score seemed very strong. Since all 7 possible objectives only deal with text data, as opposed to the wider range of applications for the more common classification models, this roBERTa model can be very useful.

Another useful State-of-the-Art model is TensorFlow Hub's Multilingual BERT Model. This variation of the BERT model is trained on multilingual Wikipedia pages. When considering the training data, it is evident that this model will have a strong understanding of general language constructs in any language, especially English. The one thing I had to keep in mind, however, is that due to this model's multilingual nature, there would be a slight possibility that the model did not have as much exposure to the English language as I hoped. However, this was proven to be an untrue assumption, as our initial exploration of this model showed fairly high levels of precision, recall, and the F1 Score (detailed further in Table C). If this experiment was conducted on all of Airbnb's global reviews, this BERT model would be useful since Airbnb is an international company with many reviews in different languages.

Chapter 3

Empirical Application

Given my motivations and topic background, this section is an overview of the Empirical Application of the technical knowledge and statistical analyses requires to answer my four research questions. My techniques are most like that of Netzer et al. (2017) in that I train multiple commonly used Machine Learning models and a few State-of-the-Art models to determine a good ensemble method for classifying Airbnb properties. However, rather than focusing on loan default classification, I focus on classifying instances of Property Misinformation.

1. Data Description

To achieve my classification objective described above, I worked with a dataset containing information on listings and reviews for San Francisco, California, Austin, Texas, Fort Lauderdale, Florida, and New York City, New York. The first three locations helped with the build and initial evaluation of the methodology, which will be described in more detail in the Methodological Development subsection.

The length of each dataset varied from city to city, with Fort Lauderdale containing 195,857 reviews, San Francisco containing 366,563 reviews, Austin containing 331,963 reviews, and New York containing 65,524 reviews. I used data from Fort Lauderdale, San Francisco, and Austin for my main analysis and data from New York to stress test my main model to assess generalizability.

While slight sampling bias does exist in my choice of cities, there are two main reasons why I chose Austin, San Francisco, and Fort Lauderdale for building my model:

- From a shortlist of 6 cities, these three locations had the largest size of data (FL: 40,776 KB, SF: 61,435 KB, TX: 64,396 KB)
- San Francisco, Austin, and Miami (45 minutes away from Fort Lauderdale) were all among the top 25 Airbnb destinations in 2016, and it looks like there was a similar trend in the early 2020s. Thus, I felt these 3 cities were representative of the most traveled locations in the Airbnb platform.

To build my methodology, I used the following information from each of the representative cities:

- Listing information: listing ID and description. The ID serves as a numerical identifier for the listing. The description information was a bit more complex, as the data I used had information such as “description” (the actual description of the property itself), “neighborhood overview”, and “transit” in separate columns within the dataset. I chose to look at the “description” column for simplicity.
- Review information: To address the problem of how often property misinformation occurs, I needed to see what guests were saying about their stay. Reviews that had more emphasis on the property itself (i.e., layout of the listing, location of the listing, amenities included, etc.) were chosen to limit the scope of the work.
- Location information: Each listing in the datasets was associated with a specific location (San Francisco, Austin, or Fort Lauderdale). This information helped in the discussion of results, as it is reasonable to assume that some locations have more examples of alleged misinformation than others.

With this data in hand, I began developing my methodology for this thesis. However, there were a few key challenges with respect to data preprocessing that I will discuss in the subsection below.

2. Methodological Challenges

Before beginning any formal exploration of the review data, I had two questions to consider when choosing locations. First, I needed to determine whether the locations were considered "popular" Airbnb destinations. This allowed me to determine whether any negative reviews for a given property in each location were simply a matter of chance or if the property itself was not a popular one among many visitors. Second, the size of the location's dataset mattered from a sampling perspective. Since I knew I would be using more robust Machine Learning methods to answer the question, a larger dataset meant a more representative sample (i.e., a better balance between regular reviews and those alleging misinformation) as well as a higher chance of better model performance.

One thing to note is that when I first started exploring the dataset, I found it difficult to find examples of misinformation, as most reviews in Airbnb are incredibly positive (BnbFacts, 2021). In addition, I found that most negative reviews spoke more about guest interactions with the host, the overall cleanliness of the space, as well as the immediate surrounding area of the listing, which do not necessarily suggest property misinformation for a given listing. Thus, I used the following approach to annotate reviews and find examples of misinformation.

I compiled all reviews for each location and created a word cloud, which gave me the most common words and phrases from all three locations. Out of all the words that were

generated, I conjectured words like “disappointing”, “horrible”, “misleading”, “deceiving”, and “inaccurate” would be useful for finding reviews alleging misinformation as well as general negative reviews. Filtering by these keywords led me to more reviews and highlight other phrases to grow my list of indicators of misinformation, such as “not as pictured” or “photoshopped”. Once I exhausted all possible indicators of misinformation, I had a working sample consisting of 1,583 rows of reviews for my methodology, with 791 reviews alleging property misinformation and 792 not making any such allegations. Each review was labeled by hand, and I used the keywords and indicators I found to label a review as “alleging misinformation” or “not alleging misinformation”.

I used a similar process to create a sample of reviews from New York City, whose dataset contained 409 reviews after my preprocessing was complete. Once I had a full working dataset, I went ahead and began experimenting with various models to answer the first and second research questions listed in this thesis.

3. Methodology

Within my methodology, I looked to examine classifications of property misinformation in three different ways, as stated earlier in this thesis: through commonly used Machine Learning models, more State-of-the-art models, and an ensemble model leveraging the strengths of the aforementioned methods. Given the nature of the problem, and the fact that the reviews needed to be annotated as “alleging misinformation” or not by hand, all the chosen methods focused on supervised Machine Learning techniques, where I gave each model a labeled sample of data and taught it to label any new samples that weren’t already labeled. One important note here is that

the preprocessing steps for each set of experiments I completed was different. When testing the more common Machine Learning models, I preprocessed the data by hand, opting to use Python's *tweet-preprocessor* and *NLTK (Natural Language Tool Kit)* packages to vectorize the review features. However, with the models that use the BERT architecture, I was able to use tokenizers that were included in the models' packages, which allowed me to reduce the manual labor that went into context-based tokenizing for sentences. Appendix B includes links to the Jupyter notebooks that show this tokenization process for both BERT-based and regular ML models.

For each experiment I completed, I chose to examine the following performance metrics: Precision, Recall, and F1-score. These metrics look to evaluate the predictions that a model generates, and they answer the following questions, respectively:

- What proportion of positive identifications was correct?
- What proportion of actual positives was identified as such?
- Given a weighted average of Precision and Recall, how well does the model *actually* perform at predicting positive predictions?

For purposes of this thesis, I considered "positive predictions" to be listings classified as "alleging misinformation".

Precision and Recall were chosen with two main assumptions in mind. First, these metrics specifically measure the frequencies of True Positive predictions and overall Positive class predictions, respectively. The problem I examined in my work was to find reviews that alleged misinformation, so looking at precision and recall allowed me to better understand how these models performed in terms of prediction quality and quantity.

In terms of the F1 score, the main assumption I had is that there was an even distribution of positive and negative classes (1 and 0). After completing a quick check of data distribution, I saw that my current dataset of 1,583 samples had an almost 1:1 ratio of reviews alleging misinformation (Label = 1) and those not alleging misinformation (Label = 0). In addition, the sklearn.metrics function *f1_score()* allowed me to easily find how well each model was doing without any manual calculations. Thus, I thought it best to include this metric in the evaluation.

A. Performance: Common Classification Techniques

With a basic understanding of how the five commonly used models work, I applied them to my data, converting the review text into token values that could be fed into each model. The initial scores for the models are shown below in Table B.

	Precision (0)	Recall (0)	F1 Score (0)	Precision (1)	Recall (1)	F1 Score (1)
Naive Bayes	75	81	78	82	77	79
Logistic Regression	76	93	84	76	75	83
LinearSVC	85	87	86	88	86	86
Random Forest	84	83	84	85	87	85
XGBoost	77	89	83	89	77	83

Table B: Precision, Recall, and F1 Score values for labels 1 (alleging misinformation) and 0 (not alleging misinformation) – Common ML models

The table shown here contains each model's scores for the performance metrics discussed earlier in this section: Precision, Recall, and the F1 score. Naïve Bayes had the lowest performance out of all 5 common Machine Learning tools chosen, with scores falling around 75-

83% for both a positive and negative prediction. One of the biggest reasons for this is that Naïve Bayes assumes Conditional Independence, where all of a model's predictors, or features, are completely independent of each other. However, this is rarely the case in real life; in the case of this paper's research questions, customer reviews are rarely fully independent, as the context of a certain keyword or key phrase matters.

On the other hand, my experiments with LinearSVC had the highest, most consistent performance for all performance metrics in each prediction class, with scores falling 85-88% across all three metrics. In Linear SVC's, the main advantage comes from the model's ability to manage many features i.e., reviews or tokenized word features. Since many of the reviews in my sample dataset contain large numbers of words, multiple instances of the same word in each dimension will allow for LinearSVC to make better classifications overall. However, I chose not to go with this model because reducing dimensionality to only focus on a few words would eliminate the context needed for a review to be fully understood i.e., a neutral-sentiment review alleging misinformation could be misclassified as not doing so since the model only focused on a few distinct words.

Overall, more common Machine Learning methods like LinearSVC and Random Forest did have some merit, as their F1 scores always fell around 80-85%. However, repeated runs showed that these models could not work very well with text data due to their tendency to handle strictly tabular, numerical data more efficiently. This issue was made clearer through my efforts to tokenize each review.

In addition, one limitation that became evident with the Grid Search Cross-validation approach is that the array of possible hyperparameter values had to be generated manually. In other words, the parameters defining a model are linked to the problem that it is trying to address

but determining the hyperparameter values to test is largely experimental, so the above models did not promise good results without extremely good intuition of what hyperparameter values work. Thus, I moved forward to explore the more State-of-the-art Machine Learning techniques I described in Chapter 2.

B. Performance: State-of-the-Art Classification Models

While the older models were a good starting point, my main intention was to use more state-of-the-art classification models to see how well they would work with predicting property misinformation. To ensure consistent results when applying each model, I decided to test both the roBERTa and Multilingual BERT models multiple times, using the average across 5 runs to populate the table. Both models performed similarly with a few differences between our chosen performance metrics, as shown in Table C.

	Precision (0)	Recall (0)	F1 Score (0)	Precision (1)	Recall (1)	F1 Score (1)
RoBERTa	88	87	88	87	88	88
RoBERTa	92	86	89	86	92	89
Multilingual BERT	90	90	90	90	90	90
Multilingual BERT	89	88	89	89	89	89

Table C: Precision, Recall, and F1 Scores for the roBERTa and BERT models

The Multilingual BERT model's scores of 88-90% across the board indicate that for all the predictions it made, it was just as good as predicting the correct label for a review. In other words, the Multilingual BERT model performed equally when predicting 1 or 0. For the roBERTa model, on the other hand, the higher recall of 92% for reviews alleging

misinformation, combined with the lower precision means that the model did predict 1 often, but the correctness of these predictions was a bit lower, which could be a possible reason the F1 score for roBERTa was slightly lower. Since both models were relatively comparable in terms of performance, I wanted to see if the differences in precision, recall, or F1 score could be accounted for with an ensemble model.

C. Ensemble Model

Overall, BERT and roBERTa's performance scores can be attributed to the individual strengths of each model. In the case of the Multilingual BERT model, the core strength here is that unlike its predecessor for creating word token vectors (i.e., Python's Word2Vec algorithm), the BERT algorithm calculates new vectors for every sentence due to the model's contextualization capabilities (Jeske, 2019). In contrast, Word2Vec uses pre-calculated vectors for each word in a document, which might result in inaccurate predictions for this task since the context of a review is what gives a specific word or sentence its meaning. BERT's biggest strength, however, is also related to one of its biggest weaknesses: a greater need for computational resources. Recalculating word token vectors for each new sentence in a document result in longer computation times, making the BERT model a bit more expensive if looking to use the model at scale.

There are a few more key weaknesses of the BERT model that roBERTa looks to address in its model architecture, including more training data (16GB vs. 160GB of data), which implies that roBERTa would be better equipped to handle larger datasets. Another major advantage of the roBERTa model is that the model looks to mask different words (a function of the model's

masked-language modeling capability) across different training iterations, resulting in the model gaining more context about certain documents (i.e., reviews) throughout the training time. However, this model did not show a performance increase due to a key weakness: roBERTa's training time is often 4-5 times larger than that of the BERT model (Khan, 2019).

Since roBERTa and BERT were relatively comparable in terms of performance, I wanted to see if the differences in precision, recall, or F1 score could be accounted for with an ensemble model. In other words, I conjectured that leveraging the strengths of these models would minimize their weaknesses.

To create my ensemble, I used two major concepts. The first concept was to keep the training and testing data the same for each model in the ensemble rather than using random splits for each dataset, as this would result in more reliability in the model. The second concept was to use a probabilistic approach in determining the final predicted label based on each model. RoBERTa and BERT both use the argmax() function to make their predictions, which takes the majority vote approach to determine the final label. In theory, a max voting approach should be helpful for an ensemble classifier to decide on the final label. However, this approach is more feasible with more than two models in an ensemble, as at least two models could possibly align in their predictions. By using the softmax() function from Scipy's special package instead, which examines the probability that a certain record (i.e., review) belongs to a certain class, I found that leveraging the probability values of a review was a more efficient way to determine the final label. Refer to Appendix B to see the code behind this logic.

Once I determined the best method to address both the concepts in my ensemble, I completed multiple testing rounds using this model on my 1,583-record review dataset. To make sure that my ensemble classifier would also work on completely new data, I also introduced a

completely new dataset of listing reviews from New York City, NY for stress testing purposes. This addition allowed me to see if the ensemble was overtrained on my original data (i.e., if the performance level was lower than expected on completely new data).

Overall, I saw better results while using the ensemble on my original data as well as on the additional data from New York City. The same trend was observed when I tested my ensemble method on my original data from San Francisco, Austin, and Fort Lauderdale, seen in Table F in the Results subsection. However, it is important to note that in both cases, the performance increase was marginal, perhaps due to the tradeoff between BERT and roBERTa's strengths being less significant than I originally thought. The reasons for the increase as well as the level of increased performance will be discussed in more detail in the section below.

4. Results of Ensemble Methodology

As seen in Table D and Table E below, the ensemble proved to have consistently higher scores on the New York Data than the individual BERT and roBERTa models. Table F shows a similar trend for my original data, but for purposes of this thesis, I will discuss the New York results in more detail, as adding this data proved that my classifier has the potential to be generalizable.

Overall, each performance metric I looked at generally scored around 87-94%, while the individual BERT models scored around 76-94% across all three runs. Out of the three performance metrics, the ones I focused most extensively on for the ensemble model were Precision and F1 Score. Not only did I want to evaluate overall model performance when it came to predicting property misinformation, but I also wanted to evaluate the correctness of positive

predictions, as accidentally flagging a perfectly normal listing could have some unwanted impacts on the legitimacy of hosts. While the range between percentages has decreased significantly with the ensemble model, the overall increase in scores was incredibly marginal.

	Precision (0)	Recall (0)	F1 Score (0)	Precision (1)	Recall (1)	F1 Score (1)
Ensemble (trial 1)	92	83	88	84	93	88
Ensemble (trial 2)	93	93	93	93	93	93
Ensemble (trial 3)	90	87	88	89	92	91

Table D: Precision, Recall, and F1 Scores for ensemble model of BERT and roBERTa on New York City Data

	Precision (0)	Recall (0)	F1 Score (0)	Precision (1)	Recall (1)	F1 Score (1)
RoBERTa	92	81	86	82	93	0.87
RoBERTa	87	92	89	93	88	91
RoBERTa	90	87	88	89	92	91
BERT	89	96	92	93	81	87
BERT	79	95	86	94	76	84
BERT	91	91	91	89	89	89

Table E: individual model scores for New York City data

	Precision (0)	Recall (0)	F1 Score (0)	Precision (1)	Recall (1)	F1 Score (1)
Ensemble (trial 1)	88	90	89	92	90	91
Ensemble (trial 2)	88	92	89	92	90	91
Ensemble (trial 3)	90	87	88	89	92	91
Ensemble (trial 4)	90	87	88	89	92	91

Table F: Precision, Recall, and F1 Scores for the ensemble model of BERT and roBERTa on San Francisco, Austin, and Fort Lauderdale data

The individual BERT and roBERTa models had an F1 score between 85-95% for predicting misinformation. Since this is a rather large interval, it was reasonable to assume that I would see a more consistent performance in my ensemble model. My assumption proved to be true when completing all trial runs for the ensemble, with F1 scores falling around 88-93%. Not only did this indicate better model performance overall, but my first research question was answered with at most 93% accuracy. In terms of Precision, both Table D and E show somewhat similar results, but the ensemble model's scores are, again, consistently scoring between 85-95%. This implied that the ensemble was better at predicting misinformation, but it went against the conjecture that the weaknesses of the individual BERT and roBERTa models would be significantly overwhelmed by their strengths in an ensemble application. The reasons for this marginal increase will be explored in the Limitations & Future work section of the paper.

Even with this smaller increase, my ensemble classifier did a much better job at labeling reviews than other experiments I completed. One reason as to why our ensemble model predicted as well as it did is that I focused on using models that were specifically designed for language-based tasks. Models like BERT and RoBERTa are attention models specifically trained to learn contextual relationships between words in each sentence or document (Horev, 2018). This contextual understanding is especially important for Airbnb reviews, which might not explicitly

allege misinformation in their reviews due to fear of a lower guest status. In addition, pretrained models already have the benefit of seeing data that is similar to the task at hand. The roBERTa model was pretrained on tweets, which can have similar sentiments to reviews. The BERT model, while not trained on sentiment-based data, can be assumed to have a better understanding of what words are more common in each language. Therefore, since BERT and RoBERTa are specifically trained on masked-language modeling, they result in better prediction performance for tasks such as misinformation detection.

Given the increased performance of this ensemble model, property misinformation occurs within Airbnb's platform more than one might assume, although the numbers imply that it is not the clearest to pinpoint without advanced Machine Learning methods. The biggest indicator for this claim is that Airbnb reviews are often neutral or positive in sentiment. Thus, reviews that allege misinformation might not focus explicitly on this phenomenon due to a guest's self-interest in keeping a good status on the platform. In addition, Airbnb reviews are overwhelmingly positive in general, so using models like my ensemble method allows for a better understanding of how certain words or phrases are used to allege property misinformation.

In terms of how to identify property misinformation, my methodology ultimately provides 3 possible options for this task: simply using the individual BERT model or roBERTa model or creating an ensemble of them. Given the similar results seen in Tables Y and Z, all three options will work well for Airbnb. However, the considerations for the rarity of property misinformation, as noted in the introduction of this thesis, warrant possible use cases for the implementation of these methods, which will be discussed in the Implications section of this paper.

Chapter 4

Implications of Findings for Airbnb

Looking at the results of my ensemble model, misinformation can be identified using more complex Machine Learning models. However, the ability to detect instances of property misinformation only solves one part of the issue, as the first two questions I looked to answer did not consider the real-world impacts that property misinformation could have on Airbnb's hosts, guests, and the platform. In this section, I focus on analyzing the properties I classified in my original dataset of San Francisco, Austin, and Fort Lauderdale reviews. More specifically, I look to uncover trends of information seen in properties that are allegedly misinforming guests. From this information, I explore the downstream consequences that misinformation can have on Airbnb's hosts and platform in two specific areas: pricing and guest booking trends.

1. Indicators of Property Misinformation

To examine what properties are more susceptible to misinformation, marketing principles such as the 4 P's need to be taken into consideration. The first of these is Price, which relates to any pricing structures associated with the product. Second, the Product dimension speaks to any information that defines the product or service offering. The Place dimension investigates where the product is sold or offered. Lastly, the Promotion aspect relates to how the product is advertised across markets. For companies like Airbnb, the host easily controls the first three dimensions. Promotion, on the other hand, can be controlled by the host, but Airbnb also plays a role in this aspect through its review structure.

Motivated by the 4 P's concept discussed above, I modeled Property Misinformation using a Logistic Regression equation to determine what property attributes are more associated with this phenomenon. The equation takes the following form:

$$\textbf{Label} = \beta_0 + \beta_i x_i + \textbf{Error term}$$

This equation looks at the label of a given data instance by providing a weighted sum of all relevant independent variables. The "Label" term refers to the presence of property misinformation, measured by a 0 (no misinformation) or 1 (alleged misinformation). The β values refer to the weights associated with each independent variable, and β_0 is the weight for the base value of 0. The term x_i refers to the independent variables I take into consideration for my regression. I examined the following independent variables for my analysis:

- Product: property types, room types, the number of people accommodated, cancellation policy specified by the host
- Place: location of the listing (San Francisco, Austin, or Fort Lauderdale)
- Price: price of the listing, the expected security deposit to ensure a booking
- Promotion: number of reviews per month, overall ratings, accuracy scores, cleanliness scores, check-in scores, host communication scores, location scores, overall value score

The code for this regression can be found in Appendix C, and the rest of this section will discuss my findings from the Logistic Regression analysis I completed in terms of the Four P's described above. The price-related variables I examined were not statistically significant variables for determining misinformation, so these will not be discussed in this section.

Variable	β value (Std. Err.)	Significance
Location: Fort Lauderdale	4.42 (1.041)	0.001
Property type: Hostel	6.20 (2.439)	0.05

Property type: Boutique Hotel	4.52 (2.149)	0.05
Property Type: Loft	4.04 (2.029)	0.05
Reviews per month	0.137 (0.0359)	0.001
Overall rating score	-0.157 (0.0188)	0.001

Table G: Weights and statistical significance of indicators affecting probability of Property Misinformation for each variable which was Significant at p < 0.05 or better

A. Product

From my analysis, I noted that the type of property could also influence misinformation allegations, especially if a guest stayed in one of these three settings: a Boutique Hotel, a Hostel, or a Loft. As shown in table G above, the coefficient estimates for these properties are as follows: 4.52, 6.20, 4.03, which indicate that properties belonging in these locations see an increase in the likelihood for property misinformation.

A Boutique Hotel property type might not indicate misinformation, as the term "Boutique" suggests a more high-end experience. However, Airbnb listings are known for the interactions that guests can have with their hosts. Thus, guests arriving to a boutique hotel might feel that property misinformation could be at play since they will need to face longer check-in processes and higher prices than regular listings. Loft and Hostel property types might be more significant indicators of misinformation because these properties are often shared with the host or with other guests. Since more guests often look for a more private experience in Airbnb, hosts that omitted the fact that the property was a hostel or a loft could see higher counts of misinformation allegations from guests, which will have an impact on their status and revenue generation as explained later.

B. Place

In terms of the listing location, Fort Lauderdale was a statistically significant location for detecting cases of property misinformation, relative to the base case of properties located in Austin. The estimated weight of this variable was 4.42, which can be interpreted in the following way: if a property is marked as being in Fort Lauderdale, the likelihood of property misinformation increases.

As a result of this finding, Airbnb would be impacted through a loss or reduction of business in this location. While most would assume that the platform's global nature allows the company to keep earnings the same, Florida's popularity as a vacation destination would cause more people to lose confidence that Airbnb can provide great stays. Thus, the reputation of the company would take some damage, as guests would choose other locations to stay.

C. Promotion

From my analysis, I noted that properties with fewer reviews per month and those with a lower overall rating (on a scale of five) were statistically significant predictors of property misinformation. When measuring the ratings per month, this variable had a weight of 0.14, and the overall ratings score had a weight of -0.156. The continuous nature of these variables results in these factors having weights with a lower magnitude than the rest of the variables I examined. However, the intuition behind the weights is the same: an increase in the ratings per month will result in a lower likelihood of property misinformation while a decrease in the overall property rating increases the likelihood of misinformation.

This phenomenon could be because guests do not have a solid understanding of the property from fewer reviews, so more aspects of the stay will feel different for them. One way that Airbnb could handle this issue is to include a message stating that "this property is a newer listing on our platform". However, this is something that can be explored in a future work.

Lower overall ratings for a given property will not only reduce the attractiveness of the listing overall, but it will also reduce the likelihood that a host would be preferred for Superhost eligibility. These impacts will be discussed in the next subsection, but they are a few of the downstream impacts that come from lower ratings from guests.

Overall, it is clear from this analysis that there exist multiple indicators of property misinformation that Airbnb and its hosts must address. This issue becomes even more important when there are financial implications that arise from misinformation. The next section will discuss what these implications could be and how Airbnb's users and business are affected financially.

2. Financial Implications of Property Misinformation

When looking at the consequences of misinformation overall on pricing, it is not enough to consider the overall price of the rental. Rather, this notion can also be extended to the number of people a host claims that their space can accommodate. It is important to note my hypothesis: properties that belong to the "alleged misinformation" group have lower prices per person, as the lower price can encourage guests to stay in a property even if they feel they only "get what they paid for". One could argue that this hypothesis might only be a result of the lack of business knowledge an Airbnb host can have about marketing in a shared economy. For example, hosts

might be more inclined to price their listing based on their own subjective views of their property rather than following a more competitive pricing method specific to the location the property is in.

I completed some further analysis to test out my hypothesis, and there exists some evidence that validates my claim about pricing, demand, and misinformation. As seen in Appendix D, when examining the median price/person across all the properties in my original training data, I found that there was an average $2.50 difference in price per number of guests a property can accommodate between properties whose descriptions differed from the guest's experience and those that aligned. More specifically, hosts that were more honest about their properties charged a median of $35/guest, as guest's standards often match the reality of the stay. However, properties marked as "alleging misinformation" by my classifier had a median price/guest of $32.50, implying that guests have the aforementioned thought of "they get what they paid for" without feeling deliberately misinformed. While the difference in prices does not seem like a large difference in theory, it can result in significant revenue loss for both Airbnb hosts and the platform itself depending on the number of guests a property can accommodate, and the number of nights guests stay at the property. This phenomenon will be discussed further in this section.

In terms of booking patterns and occupancy rates for properties, my hypothesis was similar to the claim I followed for pricing. More specifically, I hypothesized that properties that were allegedly misinforming guests saw lower occupancy rates in each period. The data I used for my analysis investigated the availability of a property within 30 days, 60 days, 90 days, and a year. However, I will use the example of occupancy rates over 30 days as the leading example for this thesis.

To examine the difference in occupancy rates across the property groups in my data, I started by creating the following variable in my dataset, which took the number of days a property was available in the past 30 days and transformed it into the number of days the property was occupied. The formula used to generate this variable is shown below:

Occupancy Rate = 30 - value(availability within 30 days)

This information is key to examining booking trends for properties that misinform their guests, as a guest has the option to cancel their booking or leave before the duration of the stay is complete, which results in lower consumer confidence and a loss of revenue for the host. Thus, occupancy rates can be a good indicator of whether guests felt that property misinformation was at play or not.

Using the occupancy variable I created, I then created a Box and Whisker plot that showed the distribution of days a given property was occupied in my dataset. This plot, along with the statistics printed with it, are shown in Appendix D. Overall, using this analysis on occupancy rates, I found that properties without property misinformation saw a median occupancy of 19 days in one month. However, those that with property misinformation saw a shorter occupancy period of 17 days in a month. Like the median differences in prices, the small difference of 2 days may not mean much on its own, but when examining the implications on misinformation on revenue, these numbers hold a much larger weight.

To illustrate the significance that property misinformation has on a host's revenues, I will take the same example of occupancy rates over 1 month. For this example, I will refer to more honest property descriptions as "good" properties, and those that misinform guests will be referred to as "bad" properties. One assumption for this example is that an average booking has 4 guests.

Good properties charge a median of price $35/guest and see a median occupancy rate of 19 days in a month. Thus, it becomes clear that hosts with good properties can charge higher prices for stays, with a total of $140/night for all four guests. In terms of revenue generated from the stay, multiplying the nightly cost by the median occupancy rate results in $2,660 of revenue for the host.

Bad properties often charge a median price of $32.50/guest and see a median occupancy of 17 nights. Using the same calculations I completed above, a booking will be priced at $130/night, $10 lower than good properties. Multiplying this price by the median occupancy for bad properties results in a total revenue of $2,210 for hosts. Clearly, the smaller differences in price per guest and occupancy rates have a ripple effect on generated revenue, as the difference between good and bad properties shows a $450 difference. In terms of percentages, this difference can be calculated as follows:

Percentage loss = average revenue loss in dollars ÷ average revenue for good properties

Using this equation, the percentage loss of revenue can be calculated as $450/$2,660, which is approximately a 17% revenue loss from alleged property misinformation. For Airbnb and its hosts alike, the implications of host revenue loss are correlated with the platform's loss of revenue. To make money from the platform's operations, Airbnb charges an earnings-based fee for hosts in which hosts give some of the profits from their stays to Airbnb. When guests allege misinformation for a given property, hosts see lower revenue, and this is a bigger problem if they invested a higher cost into the stay. For example, if Airbnb charges hosts a 9-12% fee, the company will not get as much money from a host that is associated with a misleading property. If one takes Research Question 4's findings into account, the impact is more profound on Airbnb's

earnings, but addressing the shortcomings that arise would allow for a buffer for Airbnb's operations, as explained earlier in this section.

Aside from a loss in revenue, alleged misinformation has impacts on a host's status within the Sharing Economy platform. Guests can see a host's status as well as guest reviews from previous bookings. Thus, they might be inclined to avoid hosts that have higher levels of alleged misinformation without a Superhost status. On the other hand, a guest could go ahead and book a bad property due to price competitiveness, but it is possible that they give their host a lower rating since the reality of the stay did not match their expectations.

Another implication relates to the *eligibility* of a given host for an Airbnb Superhost status. According to DPGO, a dynamic pricing tool created for short-term rental managers, eligibility for a Superhost status depends on a host's average rating, response rates, and cancellation rates ("Airbnb Plus vs Superhost," 2021). The closer each factor is to an optimal level, the more likely that hosts are eligible for Superhost status. However, if guests allege misinformation on a given property, not only will the host be pushed to adjust pricing, but they will also have lower ratings from guests, which, if below 4.8, renders a host ineligible for a Superhost status. Thus, property misinformation's downstream consequences are important to address in a platform like Airbnb since the platform treats the host's loss of business as the company's loss of business.

Chapter 5

Limitations and Future Work

While my work does provide a starting point for the topic of misinformation in the sharing economy, there are some key limitations to my work that could be better addressed through future research. The most notable of these was that my data was mostly chosen through intuition rather than a proven method. To obtain the keywords I filtered my reviews with, I first looked at various Airbnb help articles that provided examples of guest and host reviews, where I found an initial list of words like “disappointing” and “horrible”. In addition, the definition of “misinformation” includes the deliberate intent to mislead, so I conjectured reviews alleging misinformation included terms such as “misleading”, “inaccurate”, and “deceiving”. One method I used to make sure my intuition was correct with respect to my data from all three chosen cities was to compile all reviews into one sheet and generate a word cloud, which gave us the most common words found in Airbnb reviews overall as well as higher counts than I anticipated for the keywords mentioned in the Methodological Challenges subsection.

Labeling each review by hand can also be considered a limitation of this methodology because it was done based on the intuition of what counted as property misinformation instead of using a more automated tool for this job. However, since my methodology was completed with another colleague, the effects of this limitation were mitigated through using statistical tools, such as the Inter-annotator agreement measure, that ensured a similar understanding of the term “misinformation” between the two of us.

In terms of the models, one factor that contributed to the smaller increase was that the individual BERT model could only accept tokenized reviews with a sequence length of 256 tokens, which resulted in some reviews being cut off in length or chunked. This could possibly

mean that there was some information loss that could have contributed to lower performance in the model. My experiments with roBERTa proved that this issue could be fixed, but the token sequence length issue still existed even though the roBERTa model allowed for longer sequences of words to be fed into it. In addition, the longer computational times for both models, as discussed in my Ensemble Method subsection, implied that more state-of-the-art methods did require more computational resources to achieve higher levels of prediction power.

There are multiple areas in this thesis that could inspire further analysis in the future. Future work could include exploring more data, preferably international, to add to the classifier. Given that Airbnb is an international company, focusing simply on American regions would not provide a complete understanding of how property misinformation occurs as the tool cannot be deployed for the entire platform's operations. This issue becomes even more important when considering cultural determinants like language, social hierarchies, and the different ways that people (i.e., users of the Airbnb platform) interact with each other online and offline. However, a deeper dive into these determinants is not in the scope of the subject matter.

From a Data Science perspective, adding more data would not only result in a better-performing classifier, as one could leverage the multilingual capabilities of the BERT model I worked with, but the classifier would become more generalizable. In other words, examining data from different regions of the United States, or different regions of the world, would result in more concrete trends of property misinformation being uncovered by the model's predictions. For example, one could find a trend between the language semantics of Belgian guest reviews and the level of alleged property misinformation or could note that guests in New York are more aggressive with alleging misinformation than those in California.

Another possible avenue for future work on these reviews is to examine how a concept can be addressed in different ways. In my case, a listing's reviews did not always discuss misinformation in a standard way. Instead, many of the phrases that suggest misinformation are worded specifically, stating that one or more aspects of the listing, be it photos, location, amenities, or description, were inaccurate, deceiving, or misleading. Examining the specific aspect that is related to the alleged misinformation (e.g., kitchen size, number of beds, neighborhood safety) would allow researchers to understand where sharing economy platforms like Airbnb need stricter guidelines with respect to property listings, and it is possible for Airbnb to put more specific tools in place to encourage honesty in a host's listing.

Overall, this thesis was aimed at understanding how often cases of Property Misinformation were found on Airbnb and exploring the downstream effects of this phenomenon. I was able to find that property misinformation, while rare in theory, is more prevalent in practice, especially when using more complex Machine Learning models to examine this concept. While there are more effects that misinformation has on Airbnb's platform, I discussed the implications on revenue generation, as the primary aim of any business is to generate revenue for continued operations. However, my findings serve as a small contribution to the wide topic of text classification. Future work could not only answer more questions related to Airbnb, but it could potentially allow Airbnb to have a stronger foundation to address any holes in their operations.

BIBLIOGRAPHY

Airbnb Plus vs Superhost: Which is Better for Business? (2021, February 25). DPGO. https://www.dpgo.com/go/superhost-vs-airbnb-plus/

Brownlee, J. (2016, August 16). A Gentle Introduction to XGBoost for Applied Machine Learning. Machine Learning Mastery. https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/

Ghose, A., & Ipeirotis, P. G. (2006, December). Designing ranking systems for consumer reviews: The impact of review subjectivity on product sales and review quality. In Proceedings of the 16th annual workshop on information technology and systems (pp. 303-310).

Hosting on Airbnb. (n.d.). Airbnb. Retrieved March 29, 2022, from https://www.airbnb.com/hospitality

Jeske, S. (2019, November 7). Google BERT Update and What You Should Know. MarketMuse Blog. https://blog.marketmuse.com/google-bert-update/

Kowalczyk, A. (2014, October 19). Linear Kernel: Why is it recommended for text classification? SVM Tutorial. https://www.svm-tutorial.com/2014/10/svm-linear-kernel-good-text-classification/

Learn Naive Bayes Algorithm | Naive Bayes Classifier Examples. (2017, September 11). Analytics Vidhya. https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Luss, R., & d'Aspremont, A. (2015). Predicting abnormal returns from news using text classification. Quantitative Finance, 15(6), 999-1012.

Ma, B., Zhang, D., Yan, Z., & Kim, T. (2013). An LDA and synonym lexicon based approach to product feature extraction from online consumer product reviews. Journal of Electronic Commerce Research, 14(4), 304.

Oded Netzer, Alain Lemaire, and Michal Herzenstein (2017) ,"When Words Sweat: Identifying Signals For Loan Default in the Text of Loan Applications", in NA - Advances in Consumer Research Volume 45, eds. Ayelet Gneezy, Vladas Griskevicius, and Patti Williams, Duluth, MN : Association for Consumer Research, Pages: 53-56

Ph.D, S. K. (2021, May 18). BERT, RoBERTa, DistilBERT, XLNet—Which one to use? Medium. https://towardsdatascience.com/bert-roberta-distilbert-xlnet-which-one-to-use-3d5ab82ba5f8

Raj, A. (2021, January 5). The Perfect Recipe for Classification Using Logistic Regression. Medium. https://towardsdatascience.com/the-perfect-recipe-for-classification-using-logistic-regression-f8648e267592

Reviews for stays—Airbnb Help Center. (n.d.). Airbnb. Retrieved March 29, 2022, from https://www.airbnb.com/help/article/13/reviews-for-stays

Vajpayee, S. (2020, August 6). Understanding BERT — (Bidirectional Encoder Representations from Transformers). Medium. https://towardsdatascience.com/understanding-bert-bidirectional-encoder-representations-from-transformers-45ee6cd51eef

What is Logistic regression? (n.d.). Retrieved December 9, 2021, from https://www.ibm.com/topics/logistic-regression

Yiu, T. (2021, September 29). Understanding Random Forest. Medium. https://towardsdatascience.com/understanding-random-forest-58381e0602d2

17 Reasons for Bad Reviews on Airbnb. (2021, April 29). BnB Facts. https://bnbfacts.com/17-reasons-for-bad-reviews-on-airbnb/

Appendix A

Working Paper

The working paper that inspired this thesis can be found here.

Appendix B

Model Setup and Code

The code for my experiments with more common ML models can be found in this Google Colaboratory notebook. Preprocessing the data is different for the two BERT-based models, so each model's code can be found in the following Google Colaboratory notebooks:

- CardiffNLP's Twitter-based roBERTa model
- TensorFlow Hub's Multilingual BERT model

Appendix C

Logistic Regression Code

```
########################
# Logistic Regression #
########################

## Install Packages (if needed)

## Load Packages and Set Seed
set.seed(1)

## Read in Logistic Regression data
logit <- read.csv(file.choose()) ## Choose final_data.csv file

## Run Logistic Regression using GLM
logit_result <- glm(formula = Label ~ factor(Location) +
                      accommodates + factor(property_type) +
                      factor(room_type) + factor(cancellation_policy) +
                      price + security_deposit + reviews_per_month +
                      review_scores_rating, data = logit, family = "binomial")
summary(logit_result)
```

Appendix D

Statistical Analysis of Listing Price and Booking Patterns

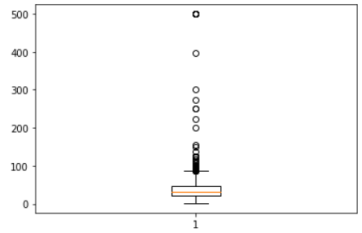
Boxplots and statistical summary of listing prices for properties marked as “1” for alleging misinformation:

```
#boxplot of price/accomodates for label = 1

data_one = data_fin[data_fin['Label'] == 1]

plt.boxplot(data_one['per_person_price'])
stats = data_one['per_person_price'].describe()
print(stats)
```

```
count    787.000000
mean      43.140575
std       51.327962
min        1.727273
25%       22.000000
50%       32.500000
75%       48.000000
max      499.500000
Name: per_person_price, dtype: float64
```

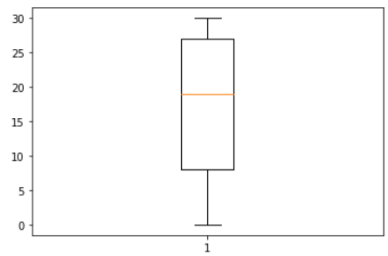


Boxplot and statistical summary of month-long booking patterns for properties marked as “0” for alleging misinformation:

```
#create boxplots for each occupancy_ variable w/ relation to label
data_z = data_occ[data_occ['Label'] == 0]

bp = plt.boxplot(data_z['occ_30'])
stats = data_z['occ_30'].describe()
print(stats)
```

```
count    598.000000
mean      17.871237
std        9.779571
min        0.000000
25%        8.000000
50%       19.000000
75%       27.000000
max       30.000000
Name: occ_30, dtype: float64
```

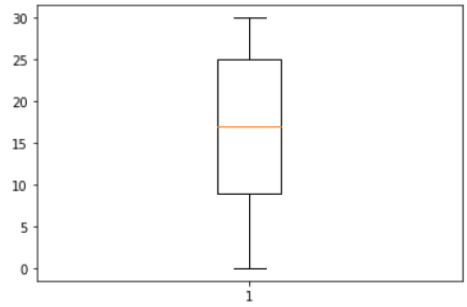


Boxplot and statistical summary of month-long booking patterns for properties marked as “1” for alleging misinformation:

```
#create boxplots for each occupancy_ variable w/ relation to label
data_o = data_occ[data_occ['Label'] == 1]

bp = plt.boxplot(data_o['occ_30'])
stats = data_o['occ_30'].describe()
print(stats)
```

```
count    787.000000
mean      16.411690
std        9.816679
min        0.000000
25%        9.000000
50%       17.000000
75%       25.000000
max       30.000000
Name: occ_30, dtype: float64
```



ACADEMIC VITA

Vaishali Devarakonda

EDUCATION

The Pennsylvania State University | Schreyer Honors College **University Park, PA**
College of Information Sciences & Technology | Bachelor of Science in Applied Data Sciences *May 2022*
Smeal College of Business | Smeal Business Fundamentals Certificate

WORK EXPERIENCE

SAP America **Newtown Square, PA**
STEM Innovators Intern *June 2018 – August 2018*

- Presented a web application and Alexa skill that predicted stock prices of IBM, Cisco, and other technology companies to the VP of the SAP North America Center of Excellence
- Guided incoming High School seniors to create a similar Alexa skill during a week-long practicum

Virtual Project Room Intern *July 2019 – January 2020*

- Managed the logging and data management of project cases in native SAP software
- Facilitated project meetings and weekly stand-ups; documented and followed up on action items
- Familiarized myself with SAP S/4HANA, SAP's Intelligent ERP system

Synchrony Financial **Stamford, CT**
Business Leadership Program Intern - Technology *June 2021 – August 2021*

- Credited as the main point of contact for optimization and rollout effort of a new third-party API design tool, SwaggerHub
- Led the onboarding efforts for 7 different teams and presented my work to more than 100 employees in the Technology and Operations department, including various Senior VPs and executive leadership

LEADERSHIP EXPERIENCE

Nittany Lion Consulting Group **University Park, PA**
Philadelphia Youth Network – Associate Consultant *August 2021 – December 2021*

- Examined current Human Resources & Payroll processes of client through conducting interviews with various employees
- Directed research efforts to benchmark various Human Resource Management systems and recommend the implementation of a new solution

Lion's Pantry – Engagement Manager *January 2022 – Present*

- Lead a team of associate consultants to recommend a new data collection methodology for a holistic understanding of Food Insecurity across the University Park campus
- Analyze Lion's Pantry's current data collection efforts through interviews with Lion's Pantry and Student Affairs staff and other Big Ten universities

PROJECTS

Case Studies **University Park, PA**
Star Textiles - PSCO *April 2021 – May 2021*

- Worked within a team of four students to analyze whether Star Textiles should move from outsourcing materials to procurement from a manufacturer
- Led the financial analysis portion of this project, where I created various Profit & Loss statements to suggest a hybrid approach of procurement

Course Projects **University Park, PA**
Student *January 2020 – December 2020*

- **Student Grade Predictor:** Performed a statistical analysis on a student grades database to determine that family relationships, social influences, and area of residence impact a student's grades in Poland
- **Grades Database:** Worked with a team of two other students to create a database to view, edit, add, or delete a student's information

MLH HackHers **New Brunswick, NJ**
Back-End Developer *February 2019*

- Created a budgeting tracker with three other students as a part of Fiserv's 2019 challenge in Rutgers University's Hackathon; Won second place with this idea
- Outlined logic for and created the regression model behind budget determination in the web application

ADDITIONAL SKILLS & INTERESTS

Programming Languages: Proficient in Python, R, Swagger, C++, SQL; experience in Java, Cypher, D3.JS
Technical Skills: Statistical Analysis methods, Data Visualization methods, Relational SQL Databases, MS Office, MS PowerBI