COMPUTATIONAL LINGUISTIC MODELS OF DECEPTIVE OPINION SPAM

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Myle Arif Ott

August 2013

COMPUTATIONAL LINGUISTIC MODELS

OF DECEPTIVE OPINION SPAM

Myle Arif Ott, Ph.D.

Cornell University 2013

Consumers increasingly rely on user-generated online reviews when making purchase decisions. However, the ease of posting reviews online, potentially anonymously, raises questions about whether unscrupulous business may be posting *deceptive opinion spam*—fraudulent or fictitious reviews that have been deliberately written to sound authentic, in order to deceive the reader. Unfortunately, as this thesis demonstrates, people are largely unable to identify deceptive opinion spam. Accordingly, it is challenging to obtain deceptive reviews for study, and, moreover, very little is known about the prevalence of deception among online reviews.

This thesis presents the first thorough investigation of deceptive opinion spam in online review communities. First, we present a novel approach for obtaining deceptive opinion spam, based on crowdsourcing, which we apply to obtain 1,280 known (gold standard) deceptive reviews of hotels and restaurants. After confirming that people are poor judges of deceptive reviews, we then present results showing that supervised Machine Learning text classifiers can be trained to detect deceptive opinion spam with nearly 90% accuracy in some settings, far surpassing human detection performance. Next, we explore linguistic features associated with deceptive reviews, and compare these features across three contextual dimensions, including the sentiment of the review (positive vs. negative), the domain of the review (hotel vs. restaurant), and the

domain expertise of the reviewer (crowdsourced workers vs. hotel employees). Finally, we present a Bayesian framework for estimating the prevalence of deception among online reviews, based on the predictions made by our Machine Learning text classifiers. Applying this framework to six online hotel review communities, we present the first empirical estimates of the rates of deception among online hotel reviews, and additionally evaluate the efficacy of increasing review posting costs to reduce the prevalence of deceptive opinion spam.

BIOGRAPHICAL SKETCH

Myle Arif Ott was born in the San Francisco Bay Area of California and later moved to Santa Monica, California, where he attended middle school. After completing the 7th grade, Myle entered the Early Entrance Program at California State University, Los Angeles. In 2006, he graduated with a Bachelor of Science degree in Computer Science, with a minor in Philosophy.

Myle entered Cornell University in 2006 and graduated the next year with a Master of Engineering degree in Computer Science under the guidance of Rich Caruana. In 2007, Myle entered Cornell's Computer Science Ph.D. program, where he studied Natural Language Processing under the guidance of Claire Cardie. He defended this thesis in July 2013.

ACKNOWLEDGEMENTS

Claire Cardie has been a wonderful advisor and mentor during my six years in the Ph.D. program at Cornell. She has taught me how to do meaningful academic research and has been unwavering in her support both through successful and unsuccessful projects. She has introduced me to the art of technical writing and has instilled in me a deep appreciation for clear and succinct communication, that will undoubtedly serve me well throughout my career.

Jeff Hancock has similarly been a valuable mentor and collaborator. He introduced me to social science research and has taught me the importance and value of interdisciplinary research. While Jeff is my Cognitive Science minor advisor, he offered me his time and guidance as though I were one of his full-time advisees, for which I am immensely appreciative.

I have also had the pleasure of collaborating with many wonderful students and researchers during my time at Cornell, including Sandeep Chawla, Yejin Choi, Aanchal Gupta, Jiwei Li, Bin Lu, Rushabh Mehta, Gabriele Piccoli, Poornima Prabhu, Stephanie Steinhardt and Sindhu Vidya. Additionally, I am thankful for having had the opportunity to know and learn from the other members of the Cornell NLP group, as well as the faculty of the Cornell Computer Science department. I am particularly thankful to Lillian Lee for all of her advice and support over the years, as well as to John Hopcroft for his valuable feedback on this dissertation and the initial proposal.

A special thanks is owed to the administrative staff in the department, particularly Michelle Eighmey, Stephanie Meik and Becky Stewart, who have kept everything organized and running smoothly during my time at Cornell and who have helped reduce many of the stresses and burdens of graduate school. I especially thank Stephanie Meik and Graeme Bailey, who were instrumental in

helping me decide to pursue my graduate studies here at Cornell.

I am also thankful to my family and friends. I especially thank my mom, brother, and girlfriend, who know me better than anyone and each of whom are appreciated for their many sacrifices, as well as their patience and support throughout my studies.

Finally, this research would not have been possible without the generous support of numerous grants and gifts, including National Science Foundation grants BCS-0624277, BCS-0904822, HSD-0624267, IIS-0968450, NSCC-0904822, and NSCC-0904913, a DARPA Deft grant, and a gift from Google. Lastly, the Jack Kent Cooke Foundation provided my tuition and expenses for my first six academic years at Cornell and connected me to a network of amazing scholars, with whom I am proud to be associated.

TABLE OF CONTENTS

Biographical Sketch . . . iii
Acknowledgements . . . iv
Table of Contents . . . vi
List of Tables . . . viii
List of Figures . . . ix

1 Introduction **1**
1.1 Opinion Spam . . . 1
1.2 Detecting Deceptive Opinion Spam . . . 5
1.3 The Effects of Context on Deceptive Opinion Spam . . . 8
1.4 Estimating the Prevalence of Deceptive Opinion Spam . . . 10
1.5 Contributions and Roadmap . . . 12

2 Obtaining Labeled Data **14**
2.1 Introduction . . . 14
2.2 Challenges and Related Approaches . . . 15
2.2.1 Traditional Approaches . . . 16
2.2.2 Non-Gold Standard Approaches . . . 18
2.3 Crowdsourcing Deception . . . 22
2.3.1 Positive Deceptive Opinion Spam of Hotels . . . 24
2.3.2 Negative Deceptive Opinion Spam of Hotels . . . 27
2.3.3 Positive Deceptive Opinion Spam of Restaurants . . . 28
2.4 Deception by Domain Experts . . . 28
2.5 Truthful Reviews . . . 29
2.5.1 Truthful Hotel Reviews . . . 30
2.5.2 Truthful Restaurant Reviews . . . 32
2.6 Data Preprocessing . . . 33
2.7 Chapter Summary . . . 34

3 A Machine Learning Approach to Detecting Deceptive Opinion Spam **35**
3.1 Introduction . . . 35
3.2 Related Work . . . 37
3.3 Human Ability to Detect Deceptive Opinion Spam . . . 38
3.3.1 Experimental Setup . . . 39
3.3.2 Results and Discussion . . . 40
3.4 Machine Learning Classifiers . . . 41
3.4.1 Naïve Bayes . . . 42
3.4.2 Support Vector Machine . . . 43
3.5 Approaches and Features . . . 43
3.5.1 Genre Identification . . . 44
3.5.2 Psycholinguistic Deception Detection . . . 44
3.5.3 Text Categorization . . . 45

3.6 Evaluation . . . 46
3.6.1 Experimental Setup . . . 46
3.6.2 Results and Discussion . . . 47
3.6.3 Linguistic Cues to Deceptive Opinion Spam . . . 50
3.7 Chapter Summary . . . 54

4 The Effects of Context on Deceptive Opinion Spam 56
4.1 Introduction . . . 56
4.2 Sentiment and Deception . . . 58
4.2.1 Human Performance . . . 59
4.2.2 Machine Learning Classifier Performance . . . 60
4.2.3 Linguistic Cues to Deception Across Sentiments . . . 63
4.3 Domain Topic and Deception . . . 65
4.3.1 Experimental Setup . . . 66
4.3.2 Results and Discussion . . . 68
4.3.3 Linguistic Cues to Deception Across Domains . . . 69
4.4 Domain Expertise and Deception . . . 71
4.4.1 Experimental Setup . . . 71
4.4.2 Results and Discussion . . . 72
4.4.3 Linguistic Cues to Deception by Domain Experts . . . 74
4.5 Discussion and Chapter Summary . . . 76

5 Estimating the Prevalence of Deceptive Opinion Spam 78
5.1 Introduction . . . 78
5.2 Related Work . . . 80
5.3 Bayesian Prevalence Model . . . 81
5.4 Signal Theory . . . 84
5.4.1 Hypotheses . . . 86
5.5 Experimental Setup . . . 86
5.5.1 Deception Classifier . . . 87
5.5.2 Classifier Sensitivity and Specificity . . . 87
5.6 Results and Discussion . . . 89
5.6.1 Assumptions and Limitations . . . 91
5.7 Chapter Summary . . . 94

6 Conclusions and Future Directions 95
6.1 Summary of Contributions . . . 95
6.2 Future Directions . . . 96

A Estimating the Prevalence of Deceptive Opinion Spam 98
A.1 Gibbs Sampler for the Bayesian Prevalence Model . . . 98
A.2 Estimating Classifier Sensitivity and Specificity . . . 99

Bibliography 101

LIST OF TABLES

2.1 Descriptive statistics for 400 positive deceptive opinion spam reviews of 20 Chicago hotels. 26
2.2 Descriptive statistics for positive Chicago hotel review data from six online review communities. 32

3.1 Deception detection performance of three human judges and two meta-judges on 160 positive Chicago hotel reviews. 40
3.2 Machine Learning deception detection performance for three approaches. 48
3.3 Top 15 highest weighted TRUTHFUL and DECEPTIVE features learned by LIWC+BIGRAMS$^{+}_{SVM}$ and LIWC$_{SVM}$. 51
3.4 A genre identification approach to detecting deceptive opinion spam. 52

4.1 Deception detection performance for three human judges and two meta-judges on 160 negative Chicago hotel reviews. 59
4.2 The effect of review sentiment on Machine Learning deception detection performance. 62
4.3 (M)eans and (S)tandard (D)eviations of four LIWC categories across 400 positive- and 400 negative-sentiment hotel reviews. . 64
4.4 SVM classifier performance on 400 RESTAURANT reviews. 67
4.5 (M)eans and (S)tandard (D)eviations of six selected LIWC categories across 800 positive HOTEL reviews and 400 positive RESTAURANT reviews. 69
4.6 Machine Learning deception detection performance across reviews from three sources: TripAdvisor, Mechanical Turk, and hotel employees. 73
4.7 (M)eans and (S)tandard (D)eviations of four selected LIWC categories across 140 positive- and 140 negative-sentiment hotel reviews from three sources: TripAdvisor, Mechanical Turk, and hotel employees. 74

5.1 Signal costs associated with six online review communities. . . . 86
5.2 Reference DECEPTIVE recall (*sensitivity*) and TRUTHFUL recall (*specificity*) of an SVM classifier trained on reviews from six communities. 88

LIST OF FIGURES

2.1 The Mechanical Turk prompt used to solicit positive deceptive opinion spam of hotels. 24
2.2 Special classes of tokens that are collapsed in preprocessing, and the special tokens with which they are replaced. 33

5.1 The Bayesian Prevalence Model in plate notation. 83
5.2 Bayesian estimates of deception prevalence versus time, for six online review communities. 90
5.3 Bayesian estimates of the prevalence of deception on TripAdvisor over time, when: (a) all reviews are included in the estimate; (b) reviews written by first-time (singleton) authors are excluded; and (c) reviews written by first- or second-time authors are excluded. 92

CHAPTER 1

INTRODUCTION

Obtaining and evaluating the opinions of others is an important step in many decision making processes. For example, people regularly seek the opinions of friends and family when faced with situations ranging from important life decisions, to choosing which businesses to patronize; politicians poll the opinions of their electorate to help inform their policies; and companies survey the opinions of their customers to discover how best to improve their offerings. And with the rise of social media, people increasingly inform their decisions with opinions and other information obtained from the Web. For example, consumers today rely on user-generated online reviews when making purchase decisions [18, 46]; politicians and political pundits look to social media to gain insights into public opinion [41, 101]; and researchers are exploring ways to use social media to predict a variety of real-world outcomes [135], including stock prices [8, 37, 110], election results [83, 122], public health [21, 92] and movie box-office revenues [3], with potential to impact business decisions in a number of industries.[1]

1.1 Opinion Spam

Unfortunately, the ease of posting content to the Web via social media, potentially anonymously, combined with the public's trust and growing reliance on opinions and other information found online, creates opportunities and incentives for abuse. Abuse of social media, or *social spam*, can include spreading misinformation and rumors [15, 68, 75, 100], inflammatory speech [127, 133],

[1]Note, however, that claims that social media can predict real-world outcomes are not without controversy; *cf.* Gayo-Avello [36] or Wong et al. [130].

blog spam [74] and social phishing [47]. This dissertation investigates social spam of online reviews of products and services, for which Jindal and Liu [48] have coined the term **opinion spam**.[2] Opinion spam can range from advertisements and promotions of unrelated products or services, to fraudulent or fake reviews that are intended to deceive the reader.

While other kinds of spam have received considerable research attention, regrettably there has been little work to date on opinion spam. Furthermore, most previous work in the area has focused on the detection of **disruptive opinion spam**—uncontroversial instances of spam that are easily identified by a human reader, for example, advertisements, questions, and other irrelevant or non-opinion text [48]. And while the presence of disruptive opinion spam is certainly a nuisance, the risk it poses to the user is minimal, since they can always choose to ignore it.

Instead, this dissertation focuses on a potentially more insidious type of opinion spam called **deceptive opinion spam**—fraudulent or fictitious opinions that have been deliberately written to sound authentic, in order to deceive the reader [88]. Unlike other kinds of spam, such as Web [14, 38] and e-mail spam [24, 119], deceptive opinion spam is neither easily ignored nor even identified by a human reader (see Section 3.3.2). For example, observe the difficulty in distinguishing between the following two hotel reviews, one of which is truthful and the other of which is deceptive opinion spam:

- "I have stayed at many hotels traveling for both business and pleasure and I can honestly stay that The James is tops. The service at the hotel is first class. The rooms are modern and very comfortable. The location is perfect within walking distance to all of the great sights and restaurants. Highly recommend to both

[2] As a matter of convenience, the terms *opinion* and *review* are used interchangeably to refer to online reviews, which are the focus of this dissertation. Nevertheless, it is our hope that the material presented here should apply not just to reviews, but to other kinds of opinions as well.

business travellers and couples."

- "My husband and I stayed at the James Chicago Hotel for our anniversary. This place is fantastic! We knew as soon as we arrived we made the right choice! The rooms are BEAUTIFUL and the staff very attentive and wonderful!! The area of the hotel is great, since I love to shop I couldn't ask for more!! We will definatly [*sic*] be back to Chicago and we will for sure be back to the James Chicago."

  Answer: see footnote.[3]

Deceptive opinion spam can be both highly influential and highly damaging, because even a small number of fake reviews can impact a business' revenue, for example, if the business' average review rating is on a rating boundary [2, 65]. Moreover, incentives to post deceptive opinion spam are growing. First, consumers increasingly rely on reviews and other information found online to make or reinforce purchase decisions [18, 46], and are reluctant to purchase a product or service offering if it has few reviews [85, 138]. Accordingly, businesses have incentive to post fake positive reviews, or ***positive* deceptive opinion spam**, to promote or hype their own offerings. On the other hand, online consumers also increasingly report changing their purchase decisions based on negative review information found online [18], and some work suggests that negative opinions may be weighted more heavily by consumers compared to positive opinions [60, 91, 116].[4] Thus, businesses also have incentive to post fake negative reviews, or ***negative* deceptive opinion spam**, to disparage or slander competitor's offerings, and to drive additional business towards their own offerings.

Unfortunately, because manual annotation of deceptive opinion spam is unreliable, there are few good sources of labeled deceptive opinion spam data for

[3]The second example review is deceptive opinion spam.

[4]Note that other work has found the consumers do not weight negative opinions more heavily than positive opinions; *cf.* Ong [85].

research. In the absence of gold standard labeled data, related studies have relied on *ad hoc* data annotation and evaluation procedures, for example, careful manual annotation, using duplicated or plagiarized reviews, or relying on noisy behavioral indicators of deception (see Section 2.2.2 for an overview). In contrast, **one contribution of this dissertation is the construction of the first large-scale, publicly available[5] corpus of *gold standard* deceptive opinion spam**, containing 1,280 deceptive reviews of hotels and restaurants, including a mix of *positive* (4- or 5-star) and *negative* (1- or 2-star) deceptive opinion spam, as well as *domain expert* deceptive opinion spam written by hotel employees.

We discuss the construction of this corpus in Chapter 2. In particular, rather than annotate existing opinions as truthful or deceptive, we instead pay people to write deceptive reviews of specific hotels and restaurants, as if they were customers. This approach has several advantages. First, it obviates the need to label the resulting data, because the opinions obtained through this approach are, by construction, deceptive opinion spam. Second, this approach mirrors one of the ways in which real-world deception occurs, for example, as in the recent case of a Belkin employee who hired people to write and post fake positive reviews for an otherwise poorly reviewed Belkin product [72].

Using this novel, gold standard corpus, **this dissertation presents the first thorough investigation of deceptive opinion spam in online review communities** through three broad questions:

1. **Detection**: Is the language of deceptive opinions different from that of truthful opinions, and if so, what are these differences, and can they be used to detect deceptive opinion spam? (Chapter 3)

2. **Context**: How is deceptive opinion spam influenced by the context of the

[5]The data used in this dissertation is available from: http://www.cs.cornell.edu/~myleott/op_spam.

deception, and how important is modeling this context when detecting deceptive opinion spam? (Chapter 4)

3. **Prevalence**: What is the prevalence of deceptive opinion spam among online reviews, what factors influence this prevalence, and what measures, if any, can be taken to reduce this prevalence? (Chapter 5)

1.2 Detecting Deceptive Opinion Spam

> "The words people use in their daily lives can reveal important aspects of their social and psychological worlds."—James W. Pennebaker et al. [97]

A person's language can be used to predict many of their hidden social and psychological attributes, for example, their gender [12, 57, 76, 112], identity [31, 69, 93, 102], location [26, 25, 43], demographics [27, 80, 103], native language [58], political orientation [19, 95, 103], personality traits [66], sentiment [64, 90], emotional state [17], and deception [73, 78, 136, 137].[6] In Chapter 3, we explore whether language can also predict, or *detect*, whether an opinion is deceptive opinion spam through three questions:

1. **Human Performance**: How effective are human judges at detecting deceptive opinion spam?

2. **Machine Learning Performance**: Can Machine Learning classifiers be trained to automatically detect deceptive opinion spam?

3. **Linguistic Cues**: What are the linguistic cues to deceptive opinion spam, and how do they relate to linguistic cues to other kinds of deception?

[6] While previous work has found that language features can sometimes predict deception [73, 78, 136, 137], it is unknown, *a priori*, whether those same techniques and language features used in previous work will apply to deceptive opinion spam.

We approach these questions through a binary classification task, in which the objective is to classify a given review as either DECEPTIVE (deceptive opinion spam) or TRUTHFUL (not deceptive opinion spam).[7] This task is additionally restricted so that classification decisions must be based only on the text of each review. This restriction is realistic, because other attributes of a review, such as the date, IP address of the poster, or number of other reviews posted by the same user, are all controlled or easily manipulated by a spammer. For example, a spammer may hire people with genuine accounts to post their deceptive opinion spam, as in the Belkin example [72], or engage in *sockpuppetry*—a social spamming technique, popular in political contexts [71, 70, 104], in which many fake online user accounts are created in order to "follow," "like," or otherwise endorse and lend credibility to a particular person, group, institution or cause.

We first evaluate the ability of human judges to detect deception, which is important for two reasons. First, there are few other baselines for this classification task; indeed, related studies [48, 73] have only considered a random guess baseline. Second, assessing human performance is necessary to validate the quality of the deceptive opinion spam corpus constructed in Chapter 2. For example, if human detection performance is found to be low, then the deceptive opinions must be convincing, and are, therefore, deserving of further attention. In general, we find that human judgements: (a) are *truth biased*, in that they are more likely to predict TRUTHFUL than DECEPTIVE; and (b) rarely predict deception better than chance, consistent with decades of traditional deception research in psychology [9].

Next, we explore whether Machine Learning and Natural Language Processing techniques can be applied to detect deceptive opinion spam. In par-

[7] This same setup has been used to study spam in other contexts, for example, e-mail, comment and Web spam (see Section 3.2), where the objective, similarly, is to classify content as SPAM or NOT-SPAM.

ticular, we recast the task as a standard *supervised* binary classification task, in which a classifier is trained by a Machine Learning algorithm on a set of labeled reviews, and is later used to classify a set of unseen test reviews. We evaluate the performance of two Machine Learning algorithms, popular in related work [48, 73, 137]—Support Vector Machine (SVM) and Naïve Bayes—and additionally compare the relative utility of three potentially complementary framings of this task. Specifically, we view this task as:

1. A standard *text categorization* task, in which classifiers are trained with content *n*-gram features [49, 114];
2. An instance of *psycholinguistic deception detection*, in which deceptive statements exemplify the psychological effects of lying, such as increased negative emotion and psychological distancing [39, 78]; and
3. A problem of *genre identification*, in which deceptive and truthful reviews are considered to be sub-genres of imaginative (fiction) and informative (non-fiction) writing, respectively [5, 105].

Using a *nested cross-validation procedure* [123] to estimate the performance of each classifier, we find that classifiers trained on features traditionally employed in: (a) psychological studies of deception and (b) genre identification, are both outperformed at statistically significant levels by a content *n*-gram–based text categorization approach. Notably, we find that classifiers trained with *n*-gram features can, in some experiments, detect deceptive opinion spam with accuracies approaching 90%, or well beyond the capabilities of human judges.

Finally, we explore possible linguistic cues to deception by deeper inspection of the best-performing Machine Learning classifiers. We find, for example, that spatial details are often predictive of TRUTHFUL reviews, suggesting that liars have difficulty encoding spatial information into their deceptions, consistent

with theories of reality monitoring and past deception research [51, 52]. However, we also find, for example, that first-person singular use is predictive of DECEPTIVE reviews, in contrast to several previous studies of deception [39, 78]. Taken together, our findings confirm the existence of linguistic cues to deceptive opinion spam, but also underscore the importance of considering the context underlying a deception, which is addressed further in Chapter 4.

1.3 The Effects of Context on Deceptive Opinion Spam

Deception has been studied in many settings, or **contexts**, with varying circumstances, stakes and motivations to deceive, as well as across modalities, such as visual, spoken, written, face-to-face and computer-mediated communication (see Vrij [124] for an overview). However, it has been challenging to identify cues to deception that apply universally across these contexts [39, 78, 124]. Indeed, some researchers now argue that cues to deception depend on the context of the deception, and therefore, approaches to detecting deception must additionally model the contextual parameters underlying each deception [7, 78, 124].

In Chapter 4, we investigate the generalizability of the Machine Learning classifiers in Chapter 3, by measuring the influence that the context of a deception has on classification performance. Specifically, we consider deceptive opinion spam across three contextual dimensions:

1. **Sentiment**: Do linguistic cues to deceptive opinion spam vary with the *sentiment* of an opinion, and if so, what influence does the sentiment of an opinion have on Machine Learning classification performance?

2. **Domain**: Do linguistic cues to deceptive opinion spam vary with the *domain*, or topic, of a review, and if so, how important is it that Machine

Learning classifiers are trained and evaluated on data of the same domain?

3. **Domain Expertise**: Do authors with expert-level domain knowledge, or *domain experts*, produce deceptive opinion spam with language that is different from that of non-experts, and if so, how can we detect deceptive opinion spam written by domain experts?

First, we consider differences between positive- and negative-sentiment deceptive opinion spam. Here, we train and test Machine Learning classifiers on reviews of opposite sentiments (OPPOSITE-SENT), and find that classification performance is significantly worse, compared to classifiers that are trained and tested on reviews of the same sentiment (SAME-SENT). However, when we train classifiers on reviews of both sentiments, jointly (JOINT-SENT), we find that performance can be competitive with the SAME-SENT classifiers. We then compare linguistic cues to deception across sentiments, and find that the biggest difference is in emotion terms; in particular, DECEPTIVE reviews seem to *exaggerate* the underlying review sentiment, relative to TRUTHFUL reviews.

Second, we explore the sensitivity of the Machine Learning classifiers, trained in Chapter 3, to the domain, or topic, of a review. Specifically, we train deception classifiers on reviews from one domain (HOTELS), and evaluate their ability to detect deceptive reviews from another domain (RESTAURANTS). We find that classification performance suffers significantly when the training and test domains differ. We also find, however, that we can leverage out-of-domain training data (e.g., HOTEL reviews), in combination with the in-domain training data (e.g., RESTAURANT reviews), to improve the accuracy of our models, compared to using just the in-domain training data.

Finally, we investigate how deceptive opinion spam is influenced by the domain expertise of the reviewer. For example, we find in Chapter 3 that deceptive

hotel reviews written by crowdsourced workers are easy to detect, in part, because they lack spatial details. Accordingly, we obtain deceptive opinion spam from hotel employees, who are *domain experts* in the hotels domain, and who have considerable spatial knowledge of their own hotels. We compare these reviews to both non-expert DECEPTIVE reviews and TRUTHFUL reviews, and find that the three kinds of reviews are linguistically distinct. Moreover, Machine Learning classifiers can distinguish between the three classes with nearly 70% accuracy, or between any two out of three classes with close to 80% accuracy.

1.4 Estimating the Prevalence of Deceptive Opinion Spam

And while detection is an important task, relatively little is known about the actual *prevalence*, or rate, of deceptive opinion spam in online review communities, or the factors that influence it. In Chapter 5, using the Machine Learning classifiers trained in Chapter 3, we investigate the prevalence of deceptive opinion spam in online review communities through three questions:

1. **Prevalence Estimation**: In the absence of gold standard annotations, can we use a noisy deception classifier to estimate the prevalence of deceptive opinion spam in a review community?
2. **Influencing Factors**: What factors influence the prevalence of deception in a review community?
3. **Prevalence Reduction**: What actions, if any, can be taken to reduce the prevalence of deception in a review community?

To answer these questions, we present a general framework for estimating the prevalence of deceptive opinion spam in an online review community, based on the Machine Learning classifiers trained in Chapter 3. Specifically, given

a noisy deception classifier that distinguishes DECEPTIVE and TRUTHFUL reviews, and inspired by studies of disease prevalence [53, 54], we present the **Bayesian Prevalence Model**—a *generative model* of deception that jointly models the classifier's uncertainty, as well as the ground-truth deceptiveness of each review. Inference for this model, which is performed via Gibbs sampling, allows us to estimate bounds on the prevalence of deception in the underlying review community, *without requiring gold standard annotations of deception*.

We then present a theoretical component to this framework, based on signaling theory from economics [117], and use it to reason about the factors that influence deception prevalence in online review communities. In particular, we argue that reviews are signals to the true, unknown quality of a product or service offering, and act to diminish the information asymmetry between past and prospective customers [44]. Accordingly, deceptive opinion spam is a *false signal*, and the prevalence of deception among online reviews is therefore a function of the *costs* and *benefits* associated with posting fake reviews.

We first consider the *benefits* associated with posting deceptive opinion spam. Here, because the spammer presumably has some vested interest in posting their review, the benefit of the review is derived directly from the existence and dissemination of that review. For example, a hotel that posts a deceptive positive review of their own property is benefited when that review is read and relied upon by a prospective customer. Moreover, this **exposure benefit** is proportional to the size of the review community's audience. Therefore, we hypothesize that deception will be more prevalent in review communities with a *high exposure benefit*, such as highly trafficked communities, compared to sites with *low exposure benefit*, for example, communities with low traffic.

We next consider the *costs* associated with posting deceptive opinion spam. Here, we consider only the **posting cost**, or the direct monetary and time costs associated with posting a review in a review community. Posting costs vary by review community, with some communities *verifying* purchases before allowing reviews to be posted, and others hiding or filtering reviews written by new or inexperienced reviewers. We hypothesize that review communities that implement these costs, i.e., communities with a *high posting cost*, will contain less deceptive opinion spam compared to communities with a *low posting cost*.

Finally, we apply the Bayesian Prevalence Model to produce the first empirical estimates of the prevalence of deception in six popular online hotel review communities: Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor and Yelp.[8] The results confirm our hypotheses that deception is most prevalent in communities with low posting costs and high exposure benefits. Moreover, we find that when actions are taken to increase a community's posting cost, for example, hiding a user's reviews until they have posted a minimum number of reviews, there are dramatic *reductions* in the estimated rates of deceptive opinion spam in that community. Lastly, we find that rates of deception in communities with a low posting cost are growing over time, approaching a maximum estimate of ~6% in August 2011, further emphasizing the importance of increased review posting costs.

1.5 Contributions and Roadmap

In Chapter 2, we introduce the first *large-scale* corpus of *gold standard* deceptive opinion spam, containing 1,280 deceptive reviews of hotels and restaurants, in-

[8]URLs for the six sites are, respectively: Expedia.com, Hotels.com, Orbitz.com, Priceline.com, TripAdvisor.com and Yelp.com.

cluding a mix of positive-sentiment (4- or 5-star) and negative-sentiment (1- or 2-star) deceptive opinion spam, as well as domain expert deceptive opinion spam written by hotel employees.

In Chapter 3, we use our novel corpus to present the first evaluation of human and Machine Learning performance at detecting gold standard deceptive opinion spam. We find that humans are poor judges of deceptive opinion spam, and rarely predict deception better than chance. In contrast, we find that Machine Learning classifiers trained on *n*-gram features can, in some experiments, detect deceptive opinion spam with accuracies approaching 90%.

In Chapter 4, we present the first in-depth analysis of the influence of several contextual factors—such as the *sentiment* of a review, the *domain* of a review, and the *domain expertise* of the reviewer—on the ability of Machine Learning classifiers to detect deceptive opinion spam. We find that while individual cues to deception vary according to the context of the review, we can train Machine Learning classifiers on reviews from several contexts to learn models of deceptive opinion spam that better generalize across contexts.

In Chapter 5, we introduce an approach for estimating the prevalence of deception in an online review community, based on the output of our Machine Learning classifiers. We apply our approach to six large-scale collections of online reviews of hotels, and present the first empirical estimates of the prevalence of deceptive opinion spam among online hotel reviews. We further use our model to study the economic incentives and reputational risks associated with posting fake reviews, and show that by increasing review posting costs, we may be able to reduce the prevalence of deceptive opinion spam.

Finally, we conclude and present directions for future work in Chapter 6.

CHAPTER 2

OBTAINING LABELED DATA

2.1 Introduction

A central challenge to studying deceptive opinion spam, and deception, in general, is that it is difficult to obtain **labeled data**, or data with reliable annotations indicating whether each message is truthful or deceptive. Previous work in the deception literature has relied on *sanctioned deception* in laboratory settings, or *unsanctioned deception* based on self-report or incentivized lying (see Section 2.2.1). In the context of deceptive opinion spam, recent work has instead relied on various non-gold standard data annotation or evaluation procedures, for example, careful *manual annotation*, *heuristic annotation*, or *unlabeled approaches* that rely on noisy behavioral indicators of deception (see Section 2.2.2).

This dissertation presents a novel approach for creating a corpus of gold standard deceptive opinion spam based on **crowdsourcing**, in which a large group of online workers are paid to write deceptive opinion spam for specific hotels and restaurants, as if they were customers (see Section 2.3). This approach has several advantages. First, data obtained in this way does not need to be labeled as DECEPTIVE or TRUTHFUL, because the crowdsourced reviews are DECEPTIVE by construction. Second, this approach is realistic, because deceptive opinion spam has been previously crowdsourced by businesses in real-world settings, for example, the Belkin case [72]. Third, this approach relies on online workers, who are easier to find in large numbers and are more representative of the general population, compared to university student participants often employed in studies of deception [4, 45, 109, 113].

Unfortunately, crowdsourcing is not appropriate for obtaining certain kinds

of data needed in this dissertation. For example, Section 4.4 requires deceptive opinion spam written by people with expert-level domain knowledge, or **domain experts**, who may not be present or may be difficult to identify in crowdsourcing marketplaces. Accordingly, we have built a novel dataset of deceptive opinion spam by soliciting reviews directly from *hotel employees*, who have expert-level domain knowledge of the HOTELS domain, in general, and of their own hotels, specifically (see Section 2.4).

Crowdsourcing is also an imperfect choice for obtaining TRUTHFUL reviews of *specific* hotels and restaurants, which are required throughout this dissertation. In particular, TRUTHFUL reviews need to be written by past customers of the hotels and restaurants of interest, and it is difficult to crowdsource workers who are qualified to write such reviews truthfully. Instead, we gather publicly available reviews found on six popular review websites, and sample from them a subset of reviews that are likely to be TRUTHFUL, based on characteristics of the user, and the hotel or restaurant (see Section 2.5).

2.2 Challenges and Related Approaches

Related work has relied on a number of approaches for obtaining labeled data to study deception. Section 2.2.1 discusses some of the approaches used in the deception literature in psychology. Section 2.2.2 discusses some of the non-gold standard approaches recently introduced by researchers specifically to study deceptive opinion spam, in the absence of gold standard labeled data.

2.2.1 Traditional Approaches

Studies of deception in the psychological literature have relied on a number of approaches for obtaining labeled deceptive data. These approaches typically fall into one of two categories, depending on whether deceptions are obtained: (a) at the direction or explicit request of a researcher, called *sanctioned lie approaches*; or (b) without any explicit instruction or permission from a researcher, called *unsanctioned lie approaches*. This section discusses some of the advantages and disadvantages of each kind of approach.

Sanctioned Lie Approaches

Lies that are told at the direction or explicit request of a researcher are called **sanctioned lies**. Traditional studies of deception often rely on sanctioned lies to obtain labeled deceptive data. For example, a researcher may ask a participant to lie about their emotional state [28]; or their personal stance on a given topic, such as abortion or the death penalty [73]; or their behavior, for example, a participant may be asked to lie after committing a mock crime [98].

In general, sanctioned lie approaches are advantageous in that they provide researchers with freedom and control over the circumstances and topic of each deception. Accordingly, sanctioned lies can be carefully solicited to allow for comparison with non-deceptive data. However, by sanctioning a lie, the researcher is implicitly endorsing the deception, which influences the psychological effects of lying [30, 35, 78]. Moreover, participants in studies of sanctioned lies are often unrepresentative of the general population, for example, when they consist primarily of university students [113].

Unsanctioned Lie Approaches

A second category of approaches rely on **unsanctioned lies**, or lies that are told without any explicit instruction or permission from the researcher. Because unsanctioned lies are not explicitly solicited, and may include real-world lies, studies of deception that rely on unsanctioned lies require additional effort to label the data, compared to studies of sanctioned lies. Unsanctioned lies can be labeled in a number of ways.

The first approach to labeling unsanctioned lies is through *self-report*, where participants are asked to identify their lies, retrospectively. This approach includes diary and survey studies, where participants are either asked to keep a diary record of their lies, or are surveyed about they lying habits. Unfortunately, while self-report methods may better capture real-world deceptions, participants can under-report deception through this approach, for example, if they fail to remember, or are embarrassed to report all of their lies. Moreover, because the labeling is done retrospectively, the data available about each lie may be minimal, unless the researcher is able to record the original communications, for example, in studies of digital deception [6, 40].

The second approach to labeling unsanctioned lies is through *incentivized lying* procedures [30, 61], where participants are not explicitly asked to lie, but are incentivized to do so in a setting where the researcher can definitively identify and label the lie. This approach is advantageous in that incentivized lies are unsanctioned, but do not rely on self-report for labeling. However, incentivized lying experiments can be challenging to setup and produce variable amounts of data, since only a fraction of participants may succumb to the incentive to lie.

Limitations

Unfortunately, the sanctioned and unsanctioned lie approaches just discussed are impractical for obtaining large-scale collections of deceptive opinion spam. First, unsanctioned lie approaches that rely on self-reports of deception are unlikely to succeed in settings where the risks of getting caught are high, for example, posting deceptive opinion spam.[1] Second, the sanctioned lie and incentivized lying approaches just discussed typically require laboratory environments, which are challenging to setup in large-scale settings, such as the Web. In Section 2.3, we present a novel sanctioned lie approach for collecting deceptive opinion spam that relies on crowdsourcing instead of a laboratory environment.

2.2.2 Non-Gold Standard Approaches

Researchers have recently proposed several approaches for studying deceptive opinion spam in the absence of gold standard deceptive data. These approaches can be broken up into three categories, depending on whether the approach relies on: (a) *manual annotation* of deceptive instances in the data; (b) *heuristic methods* for deriving approximate, but non-gold standard deception labels; or (c) *unlabeled data*, by making assumptions about the effects of deceptive behavior.

Manual Annotations

Some recent work studying deceptive opinion spam has suggested relying on **manual annotations** of deception, assigned by human judges.

Lim et al. [63] study deceptive product reviews found on Amazon.com. They develop a sophisticated software interface for manually labeling reviews as de-

[1]The Federal Trade Commission (FTC) has recently updated their guidelines on the use of endorsements and testimonials in advertising to suggest that posting deceptive opinion spam may be unlawful in the United States [29].

ceptive or truthful. The interface allows annotators to view all of each user's reviews, ranked according to dimensions potentially of importance to identifying deception, such as whether the review is duplicated, or whether the reviewer has authored many reviews of the same product group (e.g., brand) in a short period of time and with all very-high or all very-low ratings.

Wu et al. [131] study deceptive online reviews of TripAdvisor hotels by manually labeling a set of reviews according to "suspiciousness." This manually labeled dataset is then used to validate eight proposed characteristics of deceptive hotels. The proposed characteristics include features based on the number of reviews written by novice reviewers, as well as the differences between review ratings left by novice and experienced reviewers.

Li et al. [62] study deceptive product reviews found on Epinions.com. They rank reviews by user-provided helpfulness ratings, and then sample three sets of reviews from the top, middle and bottom of this ranked list. Finally, they have 10 college students manually label reviews in each sample, according to a set of tips found online for spotting fake reviews.

Mukherjee et al. [77] study groups of users who post deceptive product reviews on Amazon.com. They use a frequent itemset mining approach to select candidate "spammer groups," containing users that have posted reviews for at least three of the same products. Candidate spammer groups are then labeled by eight human judges according to a predetermined list of "spamming indicators" obtained from the Web.

Limitations Manual annotation of deception is problematic for a number of reasons. First, many of the same challenges that face manual annotation efforts in other domains also apply to annotations of deception. For example, manual annotations can be expensive to obtain, especially in large-scale settings,

such as the Web. Most seriously however, is that human ability to detect deception is notoriously poor [9]. Indeed, we find in Section 3.3 that human agreement and deception detection performance is often no better than chance; this is especially the case when considering the overtrusting nature of most human judges—a phenomenon referred to in the psychological deception literature as a *truth bias* [9, 124].

Heuristically Labeled

A second category of approaches for studying deceptive opinion spam rely on approximate, or **heuristic labels**, of deception.

Jindal and Liu [48] study the characteristics of deceptive Amazon.com reviews. They rely on an approach for heuristically labeling reviews as deceptive, based on a set of assumptions specific to their domain. In particular, after removing irrelevant content, including questions and advertisements, they label as deceptive all *duplicated* reviews, or reviews where the text heavily overlaps with the text of other reviews in the same corpus.

Feng et al. [32, 33] study deceptive reviews of hotels on TripAdvisor.com. Following the approach outlined in Chapter 3 and originally proposed in Ott et al. [88], they train Machine Learning classifiers to detect deceptive opinion spam based on the text of each review. However, they additionally consider training and testing their classifiers on heuristically-labeled reviews, which they label according to three hypotheses about deceptive hotels: (a) The average rating assigned by novice (*singleton*) reviewers will be greater than expert reviewers among deceptive hotels; (b) The ratio of strongly positive to strongly negative reviews among novice reviewers, compared to the same ratio computed among experienced reviewers, will be greater among deceptive hotels; and (c) If the

average rating among a hotel's reviews in one month is very different from the average rating of that same hotel in the previous and following months, then that hotel is more likely to be deceptive.

Limitations Heuristic labeling approaches do not produce a true gold standard corpus, but for some domains may offer an acceptable approximation. However, as with other non-gold standard approaches, certain behaviors might have other, innocent causes. For example, duplicated reviews may have been duplicated accidentally; and even if a review is duplicated maliciously, that does not mean that the original review was deceptive. Moreover, duplicated reviews are potentially better identified via traditional plagiarism or near-duplicate detection techniques [10]. Similarly, novice or singleton reviewers are not necessarily deceptive, and may post more extreme ratings for legitimate reasons. For example, someone that has an extremely great or horrible experience may be more likely to open a new account on a review website and share that experience, compared to someone who has an average or mediocre experience.

Unlabeled Approaches

A third category of approaches for studying deceptive opinion spam are **unlabeled approaches**, which rely on the effects of deceptive behavior, rather than trying to label individual messages as deceptive.

Wu et al. [132] propose a novel strategy for evaluating hypotheses about deceptive hotel reviews found on TripAdvisor, based on distortions of popularity rankings. Specifically, they test the *Proportion of Positive Singletons* and *Concentration of Positive Singletons* hypotheses of Wu et al. [131], but instead of using manually-derived labels, they test their hypotheses through the corresponding change in a hotel's ranking when these "suspicious reviews" are deleted, com-

pared to deleting random reviews.

Mayzlin et al. [67] reason about the economic incentives for posting deceptive hotel reviews by comparing a hotel's rating distribution on TripAdvisor, where anyone can create an account and post a review, to the same hotels' rating distribution on Expedia, where a verified purchase is required before a user can post a review. In particular, they apply a difference of differences approach to control for factors specific to each review community, and then explore three hypotheses relating to the incentives of various types of hotels to post deceptive opinion spam: (a) independent hotels are more likely to engage in review manipulation than branded chain hotels; (b) small owner hotels are more likely to engage in review manipulation than large owner hotels; and (c) hotels with a small management company are more likely to engage in review manipulation than hotels with a large management company.

Limitations The biggest drawback of unlabeled approaches is that they do not study deception directly, but must instead infer deception based on the effects of the deceptive behavior. Accordingly, unlabeled approaches must take great care in making assumptions about the effects of deception, in order for the results to be valid. Nevertheless, when care is taken in making these assumptions, unlabeled approaches are useful for validating hypotheses about deception, in the absence of labeled data.

2.3 Crowdsourcing Deception

This section presents a novel approach for creating a corpus of gold standard deceptive opinion spam based on **crowdsourcing**. Crowdsourcing is defined as, "the practice of obtaining needed services, ideas, or content by soliciting contri-

butions from a large group of people and especially from the online community rather than from traditional employees or suppliers" [20]. Recently, *crowdsourcing services*, such as Amazon's Mechanical Turk,[2] have gained popularity by making large-scale data annotation and collection efforts easy and financially affordable, by granting researchers access to a marketplace of anonymous online workers willing to complete small tasks.

As with traditional sanctioned lie approaches (see Section 2.2.1), crowdsourcing deception involves explicitly asking people to lie. However, while traditional sanctioned lie approaches typically rely on academic laboratory environments to obtain data, the crowdsourcing approach presented here relies on online workers from Mechanical Turk, who are easier to find in large numbers and are more representative of the general population, compared to university students [4, 45, 109, 113].

Yet, despite these advantages, crowdsourcing is not yet commonly used in the deception literature, and the work presented in this section is among the first academic applications of crowdsourcing to the task of obtaining gold standard deceptive content. Notably, Mihalcea and Strapparava [73] previously used Mechanical Turk to obtain deceptive content by asking hundreds of Mechanical Turk workers to give their true and untrue personal stances on social issues, such as abortion and the death penalty; however, their work does not consider deceptive online reviews, as investigated here.

The rest of this section is organized as follows. In Section 2.3.1, we collect 400 gold standard *positive*-sentiment (4- or 5-star) deceptive opinion spam reviews of 20 popular Chicago hotels, which we use in Chapter 3 to test the abilities of human judges and Machine Learning classifiers to detect deceptive opinion spam. In Section 2.3.2, we collect a matching set of 400 gold standard *negative*-

[2] Amazon Mechanical Turk: http://MTurk.com.

> *"Imagine you work for the marketing department of a hotel. Your boss asks you to write a fake review for the hotel (as if you were a customer) to be posted on a travel review website.* ***The review needs to sound realistic and portray the hotel in a positive light.*** *Look at their website if you are not familiar with the hotel."*

Figure 2.1: The Mechanical Turk prompt used to solicit positive deceptive opinion spam of hotels.

sentiment (1- or 2-star) deceptive opinion spam reviews of the same 20 Chicago hotels, which we use in Section 4.2 to test how the sentiment of a review affects cues to deception, and the performance of Machine Learning classifiers. In Section 2.3.3, we collect 200 gold standard positive deceptive opinion spam reviews of 10 Chicago *restaurants*, which we use in Section 4.3 to test the influence of the domain, or topic, of a deception on cues to deception, and the performance of Machine Learning classifiers.

We chose to study this data for primarily two reasons: (a) hotel and restaurant reviews are abundant in most markets, and (b) hotels and restaurants are evaluated on a relatively narrow number of aspects (unlike, for example, product reviews), so that reviews can be easily compared across different hotels or restaurants. Nevertheless, the methodology presented in this section for collecting labeled deceptive data should apply equally to collecting deceptive opinion spam in other domains, such as doctors, apartments, products, etc.

2.3.1 Positive Deceptive Opinion Spam of Hotels

Gold standard positive deceptive opinion spam is gathered from Mechanical Turk using the same procedure that we introduced in Ott et al. [88]. In particular, we create and divide 400 Mechanical Turk jobs, called

Human-Intelligence Tasks (HITs), evenly across 20 of the most popular hotels in Chicago (defined by the number of reviews on TripAdvisor), such that we obtain 20 reviews for each hotel. Popular hotels are chosen to minimize the risk of including deceptive opinion spam when building a matching sample of TRUTHFUL reviews (see Section 2.5.1). In particular, it has been hypothesized that popular offerings are less likely to become targets of deceptive opinion spam, since the relative impact of spam in such cases is small [48, 63].

The original prompt given to each worker is reproduced in Figure 2.1. Each prompt was accompanied with the name of a hotel and a link to the hotel's website. Additionally, a set of disclaimers indicated that any submission found to be of insufficient quality (e.g., written for the wrong hotel, unintelligible, unreasonably short, plagiarized,[3] etc.) would be rejected. A submission was considered unreasonably short if it contained fewer than 150 characters.

We allow workers to complete only a single HIT each, so that each review is written by a unique worker; however, because Mechanical Turk does not provide a convenient mechanism for ensuring the uniqueness of workers, we introduce a web service and associated script to implement this functionality, called Unique Turker.[4] We additionally restrict the task to workers who are located in the United States, and who maintain an approval rating on Mechanical Turk of at least 90%. Workers are allowed a maximum of 30 minutes to work on the HIT, and are paid one US dollar for an accepted submission.

It took approximately 14 days to collect 400 satisfactory deceptive opinions. Descriptive statistics appear in Table 2.1. Submissions vary quite dramatically both in length and time spent on the task. Notably, nearly 12% of the submissions were completed in *under one minute*. However, an independent two-tailed

[3]Submissions were individually checked for plagiarism at: `http://plagiarisma.net`.
[4]Unique Turker is available at: `http://uniqueturker.myleott.com`.

Table 2.1: Descriptive statistics for 400 positive deceptive opinion spam reviews of 20 Chicago hotels, gathered through Mechanical Turk. The mean hourly wage is given in terms of the harmonic mean.

	Num. Reviews	Minimum	Maximum	Mean	Sample Std. Dev.
Time spent (minutes)	400	0.08	29.78	8.06	6.32
Length (words)					
All reviews	400	25	425	115.75	61.30
Time spent < 1 minute	47	39	407	113.94	66.24
Time spent ≥ 1 minute	353	25	425	115.99	60.71
Hourly wage ($)	400	2.01	720.00	7.45	

t-test between the mean length of these submissions ($\bar{\ell}_{t<1}$) and the other submissions ($\bar{\ell}_{t\geq 1}$) shows no significant difference ($p = 0.83$). Possibly, these "quick" users started working prior to having formally accepted the HIT, in order to circumvent the imposed time limit. Indeed, the quickest submission took just 5 seconds, yet contained 114 words.

This data is used in Chapter 3 to test the abilities of human judges and Machine Learning classifiers to detect deceptive opinion spam. This data is additionally used in Chapter 4 to investigate the influence of several contextual factors, such as the *sentiment* of a review, the *domain* of a review, and the *domain expertise* of the reviewer, on the ability of Machine Learning classifiers to detect deceptive opinion spam. Finally, this data is used in Chapter 5 to train Machine Learning classifiers to estimate the prevalence of deception in online review communities.

2.3.2 Negative Deceptive Opinion Spam of Hotels

Deceptive opinion spam can also be negative in sentiment; for example, a business may post deceptive negative reviews of a competitor's offerings to drive additional business toward their own offerings. Negative deceptive opinion spam is needed in Section 4.2, along with the positive-sentiment deceptive reviews from Section 2.3.1, to determine the influence that the sentiment of a review has on cues to deception, and to evaluate the effect of sentiment on training Machine Learning classifiers to detect deceptive opinion spam.

Negative deceptive opinion spam is gathered for the same 20 Chicago hotels and in the same manner as the positive data in Section 2.3.1, and as reported in Ott et al. [87]. In the negative-sentiment setting, each HIT instructs the worker to imagine that they work for the marketing department of a hotel, as before, but now their boss asks them to *write a fake negative review of a competitor's hotel*. The other instructions and disclaimers remain identical to those used for collecting the positive-sentiment data, with one additional exception. Namely, submissions were manually inspected to ensure that they properly conveyed a negative sentiment, and approximately 2% of the submissions were discarded and replaced, where it was clear that the worker had misread the instructions and had instead written a deceptive *positive* review.

Descriptive statistics for the negative-sentiment data are similar to those of the positive-sentiment data, except that the average accepted review length was 178 words, compared to only 116 words in the positive-sentiment data. This difference in lengths across sentiments is also observed in the TRUTHFUL data where negative-sentiment reviews on TripAdvisor are on average 198 words, compared to 141 words for positive-sentiment reviews.

2.3.3 Positive Deceptive Opinion Spam of Restaurants

Section 4.3 investigates how the domain, or topic, of a deception influences cues to deception, by training Machine Learning classifiers on reviews of one domain (HOTELS), and evaluating their performance on reviews from another domain (RESTAURANTS).

Positive deceptive opinion spam reviews of restaurants are obtained from Mechanical Turk using the same procedure that was used in Section 2.3.1 to collect positive deceptive opinion spam of hotels, except that each HIT now instructs the worker to imagine that they are an employee at a restaurant, rather than a hotel. We gather 20 positive deceptive reviews for each of 10 of the most popular restaurants in Chicago, for a total of 200 positive deceptive restaurant reviews. These DECEPTIVE restaurant reviews are then paired with matching TRUTHFUL reviews in Section 2.5.2.

2.4 Deception by Domain Experts

Section 4.4 investigates how a reviewer's domain knowledge, or expertise, influences the way that they produce deceptive opinion spam. Accordingly, we need deceptive opinion spam written by people with expert-level domain knowledge, or ***domain expert* deceptive opinion spam**. Unfortunately, it is not appropriate to use crowdsourcing to obtain this data, because domain expertise is difficult to verify in crowdsourcing marketplaces. Nevertheless, small quantities of gold standard data can be obtained by identifying domain experts and soliciting deceptive reviews from them directly.

We build a novel dataset of domain expert deceptive opinion spam by soliciting reviews from *hotel employees*, who have expert-level domain knowledge

of the HOTELS domain, in general, and of their own hotels, specifically. In particular, we ask two hotel employees from each of seven hotels (14 employees total) to each write 10 deceptive positive-sentiment reviews of their own hotel, and 10 deceptive negative-sentiment reviews of their biggest local competitor's hotel. The instructions given to each hotel employee are similar to those given to Mechanical Turk workers in Sections 2.3.1 and 2.3.2. Hotel employee are each paid $50 for their participation. In total, we obtain 280 deceptive reviews of 14 hotels, including a balanced mix of positive- and negative-sentiment reviews.

Then, in order to fully study the effect of domain expertise on deception, we crowdsource an equal number of reviews for the same 14 hotels using Mechanical Turk, using the procedures presented in Section 2.3.1 (positive reviews) and Section 2.3.2 (negative reviews). Finally, corresponding TRUTHFUL reviews for these 14 hotels are obtained from TripAdvisor (see Section 2.5.1).

2.5 Truthful Reviews

In order to train supervised Machine Learning classifiers in Chapter 3, and throughout the rest of this dissertation, it is necessary to pair the DECEPTIVE reviews gathered in Sections 2.3 and 2.4 with matching TRUTHFUL reviews written by real customers. Unfortunately, while publicly available review data is ubiquitous on the Web, for the same reasons that we cannot manually annotate real-world reviews as DECEPTIVE, we also cannot manually annotate real-world reviews as TRUTHFUL.

Fortunately, recent findings, including our own findings in Chapter 5, suggest that rates of deception among travel review websites are low [67]. Moreover, we have primarily considered reviews of highly-popular[5] offerings, which

[5]Popularity is defined as the number of reviews that an offering has on TripAdvisor.

are hypothesized to contain less deceptive opinion spam, since the relative impact of spam for such offerings is small [48, 63]. We therefore gather publicly available reviews from six popular review websites, and sample from them a subset of reviews, which we label as TRUTHFUL.

We describe our TRUTHFUL hotel review data in Section 2.5.1 and our TRUTHFUL restaurant review data in Section 2.5.2.

2.5.1 Truthful Hotel Reviews

Hotel review data is obtained from six popular online hotel review communities: Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor, and Yelp. The primary data source used in this work is TripAdvisor, which is the largest review community in the HOTELS domain, by number of reviews. Reviews from the other five communities are also used, for example, to obtain sufficient numbers of negative reviews for the analysis in Section 4.2, and to test hypotheses in Chapter 5 about rates of deception in different review communities.

TripAdvisor

We mine all 64,531 reviews appearing on TripAdvisor in December 2012 from all 165 hotels in the Chicago area. This collection contains reviews for all 20 Chicago hotels for which we crowdsourced deceptive opinion spam in Sections 2.3.1 and 2.3.2. Except where specified otherwise, we filter this data by eliminating (in order):

- 4,183 non-English reviews;[6]

[6]We use the ldig social media language identification library, available here: https://github.com/shuyo/ldig.

- 1,800 reviews with fewer than 150 characters, since all crowdsourced DECEPTIVE reviews are at least 150 characters long (see Section 2.3);
- 12,315 *singleton* reviews—reviews written by new users who have not previously posted a review on TripAdvisor—since these reviews are hypothesized to be more likely to contain opinion spam [132], which would reduce the integrity of our truthful review data.

We then mine an additional 4,694 TripAdvisor reviews for the 14 hotels reviewed by hotel employees (*domain experts*) in Section 2.4.

For each of the deceptive opinion spam datasets collected in Sections 2.3 and 2.4, we sample an equal number of reviews from this TripAdvisor data, controlling for hotel and sentiment, and label them as TRUTHFUL. In order for the lengths of these TRUTHFUL reviews to be similarly distributed to those of the DECEPTIVE reviews, we sample reviews from TripAdvisor randomly, but with probabilities coming from a log-normal distribution[7] (left-truncated at 150 characters), fit to the lengths of each set of DECEPTIVE reviews.[8]

Finally, we note that there are two exceptions to this sampling process, both resulting from a shortage of *negative*-sentiment TripAdvisor reviews of our chosen hotels. First, when sampling TRUTHFUL data to match the negative deceptive opinion spam data from Section 2.3.2, we rely on a combination of the TripAdvisor data presented here and the *Six Chicago Review Communities* data presented below. Second, one of the hotels reviewed by hotel employees in Section 2.4 only has 14 negative reviews in our TripAdvisor dataset, compared to 20 negative DECEPTIVE reviews for the same hotel. Unfortunately, because we do not have access to substitute data for this hotel, we instead leave the final

[7]Work by Serrano et al. [115] suggests that a *log-normal* distribution is appropriate for modeling document lengths.

[8]We use the R package GAMLSS [108] to fit the left-truncated log-normal distribution.

Table 2.2: Descriptive statistics for *positive*-sentiment (5-star) Chicago hotel review data from six online review communities, through to August 2011.

Community	Num. Hotels	Num. Reviews
Expedia	100	4,341
Hotels.com	103	6,792
Orbitz	97	1,777
Priceline	98	4,027
TripAdvisor	104	9,602
Yelp	103	1,537

dataset slightly unbalanced.

Six Chicago Review Communities

Chapter 5 investigates how community-specific factors influence the prevalence, or rate, of deception in a review community, and therefore requires review data from multiple review communities. Accordingly, and as reported in Ott et al. [86], we obtain all Chicago hotel reviews (through to August 2011) from six popular online hotel review communities: Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor, and Yelp. Corpus statistics are given in Table 2.2.

Note that due to a bug in our data, Priceline reviews are truncated to a maximum length of 370 characters. While we ignore this artifact in the remainder of this work, the effect that this truncation may have on the corresponding results is unclear. As such, we are unable to draw strong conclusions about the Priceline review community from our data.

2.5.2 Truthful Restaurant Reviews

In Section 2.3.3, we crowdsourced positive DECEPTIVE reviews for 10 popular Chicago restaurants. Here, we mine matching positive TRUTHFUL restaurant

<URL>: Web links or other URLs.

<EMAIL>: E-mail addresses.

<TIME>: Numeric time of day, e.g., `10:30` or `23:00`.

<MONEY>: Money, e.g., `$150` or `$12.50`.

<NUMBER>: Integer or decimal numbers.

<EMOTICON>: Standard "emotion icon" character sequences, e.g., `:)` or `:(`.

<DASH>: One or more consecutive hyphen-dash (−) characters.

<HASHTAG>: A token that begins with a `#` character.

<ATMENTION>: A token that begins with a `@` character.

<ELLIPSIS>: Two or more consecutive periods.

<INTERROBANG>: Any number of consecutive exclamation points and question marks, but at least one of each.

<EXCLAMATION>: One or more consecutive exclamation points.

<QUESTION>: One or more consecutive question marks.

Figure 2.2: Special classes of tokens that are collapsed in preprocessing, and the special tokens with which they are replaced.

reviews from TripAdvisor using the same procedure used in Section 2.5.1. In particular, we mine all TripAdvisor reviews in December 2012 for the same 10 popular Chicago restaurants. After filtering reviews with fewer than 150 characters, non-English reviews, and *singleton* reviews, we are left with 3,727 reviews. From these, as before, we sample 200 reviews to label as TRUTHFUL, where reviews are sampled with probabilities based on their lengths, relative to a log-normal distribution fit to the lengths of the DECEPTIVE reviews.

2.6 Data Preprocessing

We additionally apply several preprocessing steps to all of the data that we collect in this chapter. First, we convert all reviews to ASCII, to eliminate any unusual or non-standard characters that would increase the dimentionality of

the data or cause problems in down-stream processing tasks, using a combination of BeautifulSoup[9] and Unidecode.[10] Next, we tokenize each review with the Twokenize social media text tokenizer,[11] which properly handles most URLs and emoticons. Finally, we collapse instances of several classes of tokens into special placeholder tokens, given in Figure 2.2.

2.7 Chapter Summary

We have discussed several approaches for creating and labeling deceptive content, including traditional, non-gold standard, and crowdsourced approaches. We have argued that, in the context of deceptive opinion spam, crowdsourcing approaches offer a number of benefits over traditional and non-gold standard approaches, including gold standard labeling, mirroring of real-world deception, and a worker population that better resembles the general population, compared to university students.

We have demonstrated how crowdsourcing services, such as Mechanical Turk, can be used to solicit gold standard deceptive opinion spam both quickly and cheaply. In particular, we have developed the first large-scale dataset of gold standard deceptive opinion spam, containing 1,280 deceptive reviews of hotels and restaurants, including a mix of *positive* (4- or 5-star) and *negative* (1- or 2-star) deceptive opinion spam, as well as *domain expert* deceptive opinion spam written by hotel employees. Finally, we have paired each set of DECEPTIVE reviews with matching TRUTHFUL reviews, and have made the resulting corpus publicly available.[12]

[9] http://www.crummy.com/software/BeautifulSoup/

[10] http://pypi.python.org/pypi/Unidecode

[11] Twokenize is available from: http://www.ark.cs.cmu.edu/TweetNLP.

[12] The data used in this dissertation is available from: http://www.cs.cornell.edu/~myleott/op_spam.

CHAPTER 3

A MACHINE LEARNING APPROACH TO DETECTING DECEPTIVE OPINION SPAM

3.1 Introduction

We argue in Chapter 1 that deceptive opinion spam, even in small quantities, can be both highly influential and highly damaging, for example, if the spam targets a business whose average review rating is on a rating boundary [2, 65]. Unfortunately, the massive volume of online reviews makes manual inspection and moderation of reviews impractical. Moreover, as we show in Section 3.3, people are largely unable to identify deceptive reviews. Accordingly, there is growing interest in developing *automated* approaches for identifying and filtering deceptive opinion spam.

In this chapter, we propose an approach for automatically identifying deceptive opinion spam, by training Machine Learning classifiers to distinguish between DECEPTIVE and TRUTHFUL reviews. More generally, using the gold standard data that we introduced in Chapter 2, we explore the problem of detecting deceptive opinion spam through three questions:

1. **Human Performance**: How effective are human judges at detecting deceptive opinion spam?
2. **Machine Learning Performance**: Can Machine Learning classifiers be trained to automatically detect deceptive opinion spam?
3. **Linguistic Cues**: What are the linguistic cues to deceptive opinion spam, and how do they relate to linguistic cues to other kinds of deception?

To answer the first question, and to verify the convincingness of our gold standard deceptive reviews, we ask three undergraduate students to read and

judge a subset of our data, and evaluate their ability to distinguish between DECEPTIVE and TRUTHFUL reviews (see Section 3.3). We find that human judgements: (a) are *truth biased*, in that they are more likely to predict TRUTHFUL than DECEPTIVE; and (b) rarely predict deception better than chance, consistent with decades of traditional deception research in psychology [9].

Next, we consider whether Machine Learning classifiers can be trained to automatically detect deceptive opinion spam. While we are not the first to apply Machine Learning techniques to this task (see related work in Section 3.2), our approach is the first that uses *gold standard* deceptive data. In particular, using the gold standard data described in Chapter 2, we train and evaluate two popular kinds of *supervised* Machine Learning classifiers—Support Vector Machine (SVM) and Naïve Bayes (see Section 3.4)—and compare their performance when trained on three potentially complementary sets of *features* (see Section 3.5). We find, in general, that Machine Learning classifiers significantly outperform human judgements on this task, with the best models approaching 90% accuracy in our balanced dataset.

Finally, we make several theoretical contributions by inspecting the trained Machine Learning models. In particular, we examine the largest predictors of deceptive opinion spam in each model, and compare these to linguistic cues to other kinds of deception (see Section 3.6.3). In general, our findings confirm the existence of linguistic cues to deceptive opinion spam, but also highlight the importance of considering the context underlying a deception, which we address further in Chapter 4.

3.2 Related Work

Spam has been studied in a variety of settings, for example, e-mail [24, 119], blogs [74], and the Web [14, 38, 82]. In these settings, human spam detection performance is relatively good [14], and it is possible to obtain large quantities of gold standard labeled data through manual annotation. Accordingly, research into these kinds of spam have had success using *supervised* Machine Learning techniques, which benefit from large quantities of labeled data, and which perform well in practice [119].

In contrast, researchers have only recently begun to study deceptive opinion spam, because, in part, of the challenges associated with obtaining labeled data (see the discussion in Chapter 2). For example, in the absence of gold standard deceptive reviews, Jindal and Liu [48] train Machine Learning classifiers using features based on the review text, reviewer, and product, to distinguish between *duplicate* reviews[1] (considered to be DECEPTIVE) and *non-duplicate* opinions (considered to be TRUTHFUL). Similarly, in the absence of gold standard data, Wu et al. [132] propose a strategy for detecting deceptive opinion spam based on distortions of popularity rankings. Heuristic and unlabeled approaches are unnecessary in our work, however, since we rely on *gold standard* DECEPTIVE reviews obtained through crowdsourcing (see Section 2.3).

Notably, Yoo and Gretzel [134] use a sanctioned lie approach (see Section 2.2.1) to obtain 42 gold standard deceptive opinion spam reviews, by asking tourism marketing students to each write one deceptive, positive-sentiment review of the HOUSTON AIRPORT MARRIOTT hotel. They pair these reviews with

[1]Duplicate (or near-duplicate) reviews are reviews that appear more than once in the corpus with the same (or similar) text. While these reviews may be deceptive, it is also possible for a review to be duplicated accidentally, and just because a review has been duplicated does not mean that the original review was deceptive. Moreover, such reviews are potentially detectable via off-the-shelf plagiarism detection software [10, 99].

40 reviews from TripAdvisor, and use a standard statistical test to manually compare the psychologically-relevant linguistic differences between them. In contrast, we create a much larger dataset (see Chapter 2) that we use to develop and evaluate Machine Learning deception classifiers.

Research has also been conducted on the related task of *psycholinguistic deception detection*. Mihalcea and Strapparava [73] train Machine Learning classifiers to distinguish between true and untrue views on personal issues (e.g., their stance on the death penalty). Zhou et al. [136, 137] train Machine Learning classifiers to detect deception in role-playing games designed to be played over instant messaging and e-mail. However, these studies do not consider the detection of deceptive opinion spam. Moreover, while this previous work only evaluates Machine Learning classifiers trained on a single set of features, typically n-grams, and only compares Machine Learning classifiers to one another, we evaluate and compare three sets of features (described in Section 3.5), as well as the performance of human judges (described in Section 3.3).

Lastly, automatic approaches to determining *review quality* have been studied—directly [129], and in the contexts of review helpfulness [22, 55, 84] and review credibility [128]. Unfortunately, the measures of quality employed in those works are based on human judgments, which we find in Section 3.3 to be poorly calibrated to detecting deceptive opinion spam.

3.3 Human Ability to Detect Deceptive Opinion Spam

In this section, we investigate whether human judges are capable of detecting deceptive opinion spam. Answering this question is important for several reasons. First, there are few other baselines for our classification task; indeed, related studies [48, 73] have only considered a random guess baseline. Second,

assessing human performance is necessary to validate the deceptive opinions gathered in Section 2.3. For example, if human deception detection performance is very high, then it would cast doubt on the usefulness of the crowdsourcing approach for soliciting gold standard deceptive opinion spam.

3.3.1 Experimental Setup

Our initial approach to assessing human performance on this task was with Mechanical Turk. Unfortunately, we found that some Mechanical Turk workers would select among the choices seemingly at random, presumably to maximize their hourly earnings. While a similar effect has been observed previously [1], there remains no universal solution.

Instead, we solicit the help of three volunteer undergraduate university students to make judgments on a subset of our data. This balanced subset, corresponding to the first fold of our cross-validation experiments (described in Section 3.6.1), contains 20 DECEPTIVE and 20 TRUTHFUL positive-sentiment reviews from each of four Chicago hotels (160 reviews total). Unlike the Mechanical Turk workers, our student volunteers are not offered a monetary reward. Consequently, we consider their judgements to be more honest than those obtained via Mechanical Turk.

We evaluate the performance of the human judges both individually and collectively. Specifically, we report each judge's accuracy, and TRUTHFUL and DECEPTIVE (P)recision, (R)ecall and $(F)_1$-score.[2] Additionally, to test the extent to which the individual human judges are biased, we consider two meta-judges: (a) a MAJORITY meta-judge, that predicts DECEPTIVE whenever at least two out of three human judges believe the review to be deceptive; and (b) a SKEPTIC

[2] F_1-score is the harmonic mean of precision and recall.

Table 3.1: Deception detection performance of three human judges and two meta-judges on 160 reviews, corresponding to the first fold of our cross-validation experiments in Section 3.6.1. Boldface denotes the largest value in each column.

		TRUTHFUL			DECEPTIVE		
	Accuracy	**P**	**R**	**F**	**P**	**R**	**F**
JUDGE 1	**61.9**	57.9	87.5	**69.7**	74.4	36.3	48.7
JUDGE 2	56.9	53.9	**95.0**	68.8	**78.9**	18.8	30.3
JUDGE 3	53.1	52.3	70.0	59.9	54.7	36.3	43.6
MAJORITY	58.1	54.8	92.5	68.8	76.0	23.8	36.2
SKEPTIC	60.6	**60.8**	60.0	60.4	60.5	**61.3**	**60.9**

meta-judge, that predicts DECEPTIVE whenever *any* human judge believes the review to be deceptive.

3.3.2 Results and Discussion

Human and meta-judge performance is given in Table 3.1. It is clear from the results that human judges are not particularly effective at this task. Indeed, a two-tailed binomial test fails to reject the null hypothesis that JUDGE 2 and JUDGE 3 perform at-chance ($p = 0.003, 0.10, 0.48$ for the three judges, respectively). Inter-annotator agreement scores are similarly low for the three judges. For example, Fleiss' kappa computed among the three judges is 0.11, corresponding to only *slight agreement* between annotators [59]. Furthermore, the largest pairwise Cohen's kappa is 0.12, between JUDGE 2 and JUDGE 3—a value far below generally accepted pairwise agreement levels. We suspect that agreement among our human judges is so low *precisely because* humans are poor judges of deception, and therefore perform nearly at-chance relative to one another.

The finding that humans are poor judges of deception is also well supported in the deception literature in psychology. For example, untrained humans often

focus on unreliable cues to deception [124], such as increased usage of second-person pronouns [121]. Moreover, a recent meta-analysis of 206 studies found that humans rarely predict deception better than chance [9].

Truth Bias We also observe that all three human judges suffer from a *truth bias*, a common finding in deception detection research in which people are more likely to classify something as truthful than deceptive [9, 124]. For example, one of the human judges (JUDGE 2) classified fewer than 12% of the reviews as DECEPTIVE. We note that this bias is effectively smoothed by the SKEPTIC meta-judge, which produces nearly perfectly class-balanced predictions.

One possible explanation of the observed truth bias is that our human judges were not told the class-balance, i.e., they did not know that the dataset contained 50% DECEPTIVE and 50% TRUTHFUL reviews. While this constraint is realistic, to see the impact that such knowledge would have on this task, we re-evaluated human performance on this same data with three new undergraduate students, who this time were made aware of the class-balance in advance. We found that detection performance remained similar to that given in Table 3.1, with accuracies for the three distribution-aware human judges of 53.1%, 57.5% and 66.9%. However, this second batch of human judgments were far less truth-biased, with the three judges annotating between 33% and 52% of reviews as DECEPTIVE, compared to between 12% and 33% among the original judges.

3.4 Machine Learning Classifiers

Following previous work [73, 137], we train and compare two kinds of Machine Learning classifiers on our deception detection task: Naïve Bayes (Section 3.4.1) and Support Vector Machine (Section 3.4.2). We train these classifiers in a *su-*

pervised manner, in which each classifier is given a set of labeled *training data*, containing pairs of reviews and their associated labels. We represent each review as a feature vector, $\vec{x_i} \in \mathbb{R}^{|V|}$, where $|V|$ corresponds to the size of the feature space, and each label as a binary indicator, $y_i \in \{-1, 1\}$, where -1 corresponds to TRUTHFUL and 1 corresponds to DECEPTIVE. This section gives an overview of each kind of classifier, with an emphasis on how they make predictions.

3.4.1 Naïve Bayes

Naïve Bayes classifiers rely on Bayes' rule to make classification decisions. In our setting, this means that for each unseen test review, $\vec{x}$, the Naïve Bayes classifier will predict a label, $\hat{y} \in \{\text{DECEPTIVE}, \text{TRUTHFUL}\}$, according to the following decision rule:

$$\begin{aligned} \hat{y} &= \arg\max_c P_\theta(y = c \mid \vec{x}) \\ &\propto \arg\max_c P_\theta(y = c) \cdot P_\theta(\vec{x} \mid y = c), \end{aligned} \tag{3.1}$$

where parameters, θ, are estimated from the training data. Specifically, $P_\theta(y = c)$ corresponds to the *class prior*, or the probability of seeing class c in the training data, and $P_\theta(\vec{x} \mid y = c)$ corresponds to the *likelihood*, or the probability of seeing the review, $\vec{x}$, conditioned on the class of the review being c. Furthermore, when the class prior is *uniform*, for example, when the classes are balanced (as in our case), we can simplify (3.1) to the maximum likelihood classifier [94]:

$$\hat{y} = \arg\max_c P_\theta(\vec{x} \mid y = c) \tag{3.2}$$

Under (3.2), both the Naïve Bayes classifier used by Mihalcea and Strapparava [73] and the language model classifier used by Zhou et al. [137] are equivalent. In particular, for the n-gram features described below (see Section 3.5.3),

we can estimate individual *language models*, $P_\theta(\vec{x} \mid y = c)$, for the DECEPTIVE and TRUTHFUL training data, and compare the scores from those models directly to make predictions. Following Zhou et al. [137], we estimate language modules with the SRI Language Modeling Toolkit [118]. We additionally smooth the models using the interpolated Kneser-Ney method [16], which has been shown to closely approximate hierarchical Pitman-Yor language models [120].

3.4.2 Support Vector Machine

Support Vector Machine (SVM) classifiers find a high-dimensional separating hyperplane between two groups of data, and have been found to perform well on textual data [49, 73, 137]. To simplify our feature analysis in Section 3.6.3, we restrict our evaluation to *linear* SVMs, which learn a weight vector $\vec{w}$ and bias term b, such that an unseen test review, $\vec{x}$, can be classified by the decision rule:

$$\hat{y} = sign(\vec{w} \cdot \vec{x} + b) \tag{3.3}$$

We use SVM^{light} [50] to train our linear SVM classifiers, and tune the parameter, C, through cross-validation (see Section 3.6.1). Following standard practice, we additionally normalize feature vectors to unit-length.

3.5 Approaches and Features

To obtain a deeper understanding of the nature of deceptive opinion spam, we explore the relative utility of three potentially complementary framings of this task. Each framing corresponds to a set of *features*, which we use to train the Machine Learning classifiers in Section 3.4. Specifically, we view this task as:

1. A **genre identification** task, in which DECEPTIVE and TRUTHFUL reviews

are sub-genres of imaginative (fiction) and informative (non-fiction) writing, respectively [5, 105]; these classifiers are trained on *syntactic features*.

2. A **psycholinguistic deception detection** task, in which deceptive statements exemplify the psychological effects of lying, such as increased negative emotion and psychological distancing [39, 78]; these classifiers are trained on *dictionary-based features*.

3. A standard **text categorization** task, in which DECEPTIVE and TRUTHFUL reviews represent distinct document categories [49, 114]; these classifiers are trained on *content features*.

The rest of this section discusses the details of each of these three feature sets.

3.5.1 Genre Identification

Work in computational linguistics has shown that the frequency distribution of parts-of-speech (POS) in a text are often dependent on the genre of the text [5, 105]. In our *genre identification* approach to detecting deceptive opinion spam, we test if such a relationship exists for DECEPTIVE and TRUTHFUL reviews by constructing features containing the frequencies of each POS, obtained using the Stanford Parser [56]. Our expectation is that DECEPTIVE reviews will better resemble fiction, or imaginative writing, while TRUTHFUL reviews will better resemble non-fiction, or informative writing. These features are also intended to provide a good baseline with which to compare our other approaches.

3.5.2 Psycholinguistic Deception Detection

In our *psycholinguistic deception detection* approach, we expect DECEPTIVE reviews to exemplify the psychological effects of lying, such as increased neg-

ative emotion and psychological distancing [39, 78]. In particular, we create psychologically-relevant features using the *Linguistic Inquiry and Word Count* (LIWC) software [96]—an automated text analysis tool, popular in the social sciences, that has been used to detect personality traits [66], study tutoring dynamics [13], and analyze deception [39, 73, 126].

LIWC counts and groups the number of instances of nearly 4,500 keywords into 80 psychologically meaningful dimensions. We construct one feature for each of the 80 LIWC dimensions, which can be summarized broadly under the following four categories:[3]

1. **Linguistic processes**: Functional aspects of text, for example, the average number of words per sentence, the rate of misspellings, swear words, etc.
2. **Psychological processes**: Includes all social, emotional, cognitive, perceptual and biological processes, as well as anything related to time or space.
3. **Personal concerns**: Any references to work, leisure, money, religion, etc.
4. **Spoken categories**: Primarily filler and agreement words.

While other psycholinguistic feature categories have been considered in past deception detection work, notably those of Zhou et al. [136], early experiments found the LIWC features to perform best. Indeed, the LIWC2007 software used in this work subsumes most of the categories used in this previous work.

3.5.3 Text Categorization

In contrast to the other two approaches just discussed, our *text categorization* approach to detecting deception allows us to model both the content and shallow context of each review through *n*-gram features. Specifically, an *n*-gram feature

[3]More details about LIWC and the LIWC categories are available at: `http://liwc.net`.

representation of a review containing W words will produce $(W-n+1)$ features, corresponding to sequential word-level windows of size n. For example, we might represent the text "`my wife and I`" with four 1-grams, or *unigrams* (`my` | `wife` | `and` | `i`); three 2-grams, or *bigrams* (`my_wife` | `wife_and` | `and_i`); or two 3-grams, or *trigrams* (`my_wife_and` | `wife_and_i`).

We consider three n-gram feature sets, with the corresponding features lowercased and unstemmed: UNIGRAMS (1-grams), BIGRAMS$^+$ (1- and 2-grams) and TRIGRAMS$^+$ (1-, 2- and 3-grams), where the superscript ($^+$) is used to indicate that higher-order n-gram feature sets subsume the lower-order ones.

3.6 Evaluation

3.6.1 Experimental Setup

We evaluate the classifiers and features described in Sections 3.4 and 3.5, respectively, using a 5-fold *nested* cross-validation (CV) procedure [123], where model parameters are selected for each test fold based on nested CV experiments on the training folds. Each of the five folds contains 20 DECEPTIVE (see Section 2.3.1) and 20 TRUTHFUL (see Section 2.5.1) *positive*-sentiment reviews of four Chicago hotels (800 reviews total). Because folds contain reviews of distinct hotels, learned models are always evaluated on reviews from unseen hotels. We explore classification of *negative*-sentiment hotel reviews in Chapter 4.

For each combination of classifier and features, we report the (P)recision, (R)ecall and $(F)_1$-score, which we compute using a micro-average, i.e., from the *aggregate* true positive, false positive and false negative rates, as suggested by Forman and Scholz [34]. Note that we only report performance for Naïve Bayes classifiers (Section 3.4.1) using the *text categorization* feature set (Section 3.5.3).

While we tried other combinations of features with Naïve Bayes classifiers, they consistently underperformed the corresponding Support Vector Machine classifiers, and the results are therefore excluded for brevity. The superior performance of Support Vector Machine over Naïve Bayes classifiers on this task is discussed further in Section 3.6.2.

3.6.2 Results and Discussion

Comparison to Human Performance Results appear in Table 3.2. We observe that automated classifiers outperform human judges for every metric, except truthful recall, where JUDGE 2 performs best. However, as we mention in Section 3.3, JUDGE 2 classified fewer than 12% of opinions as deceptive. Thus, while achieving 95% truthful recall, this judge's corresponding precision was not significantly different from chance (two-tailed binomial $p = 0.4$).

Genre Identification Approach Among the automated classifiers, baseline performance is given by the simple genre identification approach (POS_{SVM}), proposed in Section 3.5.1. We find that even this simple automated classifier outperforms most human judges (one-tailed sign test $p = 0.06, 0.01, 0.001$ for the three judges, respectively, on the first fold). This result is best explained by theories of reality monitoring [51], which suggest that truthful and deceptive opinions might be classified into informative and imaginative genres, respectively. We explore this relationship further in Section 3.6.3.

Psycholinguistic and Text Categorization Approaches Both remaining automated approaches to detecting deceptive opinion spam outperform the simple genre identification baseline just discussed. Specifically, the psycholinguistic

Table 3.2: Machine Learning classifier performance for three approaches based on nested 5-fold cross-validation experiments and 800 positive-sentiment Chicago reviews. Specific classifiers are indicated via subscripts. Human performance is repeated here for JUDGE 1, JUDGE 2 and the SKEPTIC meta-judge, although they cannot be compared directly, since the 160-review subset on which human performance was assessed corresponds to only a single cross-validation fold (see Section 3.3).

			TRUTHFUL			DECEPTIVE		
Approach	**Features**	**Accuracy**	**P**	**R**	**F**	**P**	**R**	**F**
Genre identification	POS_{SVM}	73.0	75.3	68.5	71.7	71.1	77.5	74.2
Psycholinguistic deception detection	LIWC_{SVM}	76.8	77.2	76.0	76.6	76.4	77.5	76.9
Text categorization	$\text{UNIGRAMS}_{\text{SVM}}$	88.4	89.9	86.5	88.2	87.0	90.3	88.6
	$\text{BIGRAMS}^{+}_{\text{SVM}}$	89.6	90.1	89.0	89.6	89.1	90.3	89.7
	$\text{LIWC+BIGRAMS}^{+}_{\text{SVM}}$	**89.8**	89.8	**89.8**	**89.8**	**89.8**	89.8	**89.8**
	$\text{TRIGRAMS}^{+}_{\text{SVM}}$	89.0	89.0	89.0	89.0	89.0	89.0	89.0
	$\text{UNIGRAMS}_{\text{NB}}$	88.4	**92.5**	83.5	87.8	85.0	**93.3**	88.9
	$\text{BIGRAMS}^{+}_{\text{NB}}$	88.9	89.8	87.8	88.7	88.0	90.0	89.0
	$\text{TRIGRAMS}^{+}_{\text{NB}}$	87.6	87.7	87.5	87.6	87.5	87.8	87.6
Human	JUDGE 1	61.9	57.9	87.5	69.7	74.4	36.3	48.7
	JUDGE 2	56.9	53.9	95.0	68.8	78.9	18.8	30.3
	SKEPTIC	60.6	60.8	60.0	60.4	60.5	61.3	60.9

deception detection approach (LIWC_{SVM}), proposed in Section 3.5.2, performs 3.8% more accurately (one-tailed sign test $p = 0.02$), and the text categorization approach, proposed in Section 3.5.3, performs between 14.6% and 16.6% more accurately. However, best performance overall is achieved by combining features from these two approaches. Particularly, the combined model $\text{LIWC+BIGRAMS}^{+}_{\text{SVM}}$ is 89.8% accurate at detecting deceptive opinion spam, although the result is not significantly better than $\text{BIGRAMS}^{+}_{\text{SVM}}$ alone.

We further observe that models trained on UNIGRAMS—the simplest *n*-gram feature set—outperform all non–text-categorization approaches, and models trained on BIGRAMS^{+} perform *even better* (one-tailed sign test $p = 0.07$). This suggests that a universal set of keyword-based deception cues (e.g., LIWC) is not the best approach for detecting deceptive opinion spam, and a more detailed set of features (e.g., content *n*-grams) might be necessary to achieve state-of-the-art deception detection performance. We explore this further in Section 3.6.3.

Machine Learning Classifiers Finally, and in contrast to related deception detection work [73, 137], we observe that SVM classifiers nearly always outperform NB classifiers at this task, although not significantly ($p = 0.50, 0.28, 0.23$ for UNIGRAMS, BIGRAMS^{+}, and TRIGRAMS^{+}, respectively). Previous work by Ng and Jordan [79] has found that generative approaches, such as NB, only outperform discriminative approaches, such as SVM, when the training set size is small. Therefore, our findings may be explained by our large training set size, relative to previous work.

We note that we also evaluated the performance of Logistic Regression (LR) classifiers on this task, which are related to SVM classifiers, and are popular in related work [48, 77, 78, 136]. However, we observed that SVM classifiers nearly always outperform LR classifiers on our binary classification task, and we have

therefore reported only SVM performance. We note further that LR classifiers are often preferred when output probabilities are desired, rather than just binary classification decisions. While LR classifiers more readily produce these kinds of probabilities, it is also possible to obtain probabilities by calibrating the output of SVM classifiers [81].

3.6.3 Linguistic Cues to Deceptive Opinion Spam

Machine Learning classifiers significantly outperform human performance on this task. To better understand what these classifiers have learned, and any corresponding linguistic cues to deception, we can examine the weight vector, $\vec{w}$, learned by the linear SVM classifiers.

Informative and Imaginative Genres Work by Rayson et al. [105] has found strong distributional differences between informative and imaginative writing, namely that the former typically consists of more nouns, adjectives, prepositions, determiners, and coordinating conjunctions, while the latter consists of more verbs,[4] adverbs,[5] pronouns, and pre-determiners. Indeed, we find that the weights learned by POS_{SVM} (found in Table 3.4) are largely in agreement with these findings, notably except for adjective and adverb *superlatives*, the latter of which was also found to be an exception by Rayson et al. [105]. However, that deceptive opinions contain more superlatives is expected, since deceptive writing (but not necessarily imaginative writing, in general) often contains exaggerated language [11, 39]. In fact, as we will show in Section 4.2.3, DECEPTIVE reviews tend to exaggerate the underlying review sentiment, both in the case of positive-sentiment and negative-sentiment reviews.

[4] *Past participle* verbs were an exception.
[5] *Superlative* adverbs were an exception.

Table 3.3: Top 15 highest weighted TRUTHFUL and DECEPTIVE features learned by LIWC+BIGRAMS$^{+}_{SVM}$ and LIWC$_{SVM}$. Ambiguous features are subscripted to indicate the source of the feature. LIWC features correspond to groups of keywords as explained in Section 3.5.2.

LIWC+BIGRAMS$^{+}_{SVM}$		LIWC$_{SVM}$	
TRUTHFUL	DECEPTIVE	TRUTHFUL	DECEPTIVE
<dash>	chicago	hear	i
<ellipsis>	my	number	family
on	hotel	allpunct	ppron
location	,_and	negemo	see
)	luxury	dash	pronoun
allpunct$_{LIWC}$	experience	exclusive	leisure
floor	hilton	we	exclampunct
(	business	sexual	sixletters
the_hotel	vacation	period	posemo
bathroom	i$_{BIGRAMS^{+}}$	otherpunct	comma
small	spa	space	cause
helpful	looking	human	auxverb
<money>	while	past	future
hotel_.	husband	inhibition	perceptual
other	my_husband	assent	feel

Table 3.4: Average feature weights learned by POS_{SVM}. Based on work by Rayson et al. [105], we expect weights on the left to be negative (predictive of TRUTHFUL reviews), and weights on the right to be positive (predictive of DECEPTIVE reviews). Boldface entries are at odds with these expectations.

Informative (TRUTHFUL)			Imaginative (DECEPTIVE)		
Category	**Type**	**Weight**	**Category**	**Type**	**Weight**
Nouns	singular	−0.008	Verbs	base	0.057
	plural	−0.002		past tense	**−0.041**
	proper, singular	**0.041**		present participle	0.089
	proper, plural	−0.091		singular, present	0.031
Adjectives	general	−0.002		third person	**−0.026**
	comparative	−0.058		singular, present	
	superlative	**0.164**		modal	0.063
Prepositions	general	−0.064	Adverbs	general	**−0.001**
Determiners	general	−0.009		comparative	0.035
Coord. conj.	general	−0.094	Pronouns	personal	0.098
Verbs	past participle	−0.053		possessive	0.303
Adverbs	superlative	**0.094**	Pre-determiners	general	**−0.017**

Spatial Details Table 3.3 gives the top 15 highest weighted features learned by LIWC+BIGRAMS$^{+}_{\text{SVM}}$ and LIWC$_{\text{SVM}}$ for each class (DECEPTIVE and TRUTHFUL). We observe that TRUTHFUL reviews, compared to DECEPTIVE reviews, tend to include more sensorial and concrete language, and include more specific spatial details (e.g., `small`, `bathroom`, `on`, `floor`, `location`). These findings are in agreement with theories of reality monitoring [51], and recent work by Vrij et al. [125], which suggests that liars have considerable difficultly encoding spatial information into their lies. In contrast, we find that DECEPTIVE reviews generally focus on general features of the hotel (e.g., `luxury`, `spa`), or aspects external to the hotel (e.g., `husband`, `business`, `vacation`), possibly because the authors were crowdsourced from Mechanical Turk, and had no prior experience with the hotels they were reviewing.

Sentiment We also acknowledge several findings that, on the surface, are in contrast to previous studies of deception. For instance, while deception is often associated with increased negative emotion terms [39, 78], our LIWC$_{\text{SVM}}$ classifier found positive emotion terms (posemo$_{\text{LIWC}}$) to be predictive of DECEPTIVE reviews, and negative emotion terms (negemo$_{\text{LIWC}}$) to be predictive of TRUTHFUL reviews. This result can be explained, however, as our deceivers exaggerating the underlying positive sentiment of their reviews. We explore the relationship between sentiment and cues to deception further in Section 4.2.

First-Person Singular Pronouns Deception has also previously been associated with decreased usage of first-person singular pronouns, an effect attributed to *psychological distancing*, whereby deceivers talk less about themselves due either to a lack of personal experience, or to detach themselves from the lie [78, 136]. In contrast, we find increased first person singular (i$_{\text{LIWC}}$) to be

among the largest indicators of deception. We suspect that this relates to an effect observed in previous studies of deception, where liars inadvertently undermine their lies by overemphasizing those aspects of their deception that they believe reflect credibility [9, 23]. Accordingly, because our deceivers were instructed to produce a realistic deceptive review from the perspective of a customer, they may have overemphasized their own presence, because they believed that doing so was necessary for their review to be perceived as credible.

Context Finally, we found in Section 3.6.2 that classifiers trained on *n*-gram features significantly outperformed classifiers trained on more abstract features, such as LIWC or parts-of-speech. Indeed, we observe that many of the top *n*-gram features for the DECEPTIVE class are highly specific to the context of the training data (e.g, `chicago`, `hotel`, `hilton`), suggesting that these features are important to obtaining optimal classification performance. Unfortunately, by relying on these highly context-specific features, it is not clear how well our *n*-gram classifiers will generalize to other contexts. We further explore the effect that the context of a deception has on detection performance in Chapter 4.

3.7 Chapter Summary

In this chapter, we have studied the first large-scale dataset containing *gold standard* deceptive opinion spam. In particular, we have shown that the detection of deceptive opinion spam is well beyond the capabilities of human judges, most of whom perform roughly at-chance. Accordingly, we have introduced three *automated* approaches to deceptive opinion spam detection, based on insights coming from research in computational linguistics and psychology. We find that while standard *n*-gram–based text categorization is the best individual de-

tection approach, a *combination* approach using psycholinguistically-motivated features and *n*-gram features can perform slightly better.

Finally, we have made several theoretical contributions. Specifically, our findings suggest the importance of considering the context underlying a deception (e.g., content *n*-gram features), rather than relying on coarse deception cues (e.g., LIWC). We have also presented results based on the feature weights learned by our classifiers that illustrate the difficulties faced by liars in encoding spatial information. Lastly, we have discovered a plausible relationship between deceptive opinion spam and imaginative writing, based on POS distributional similarities.

CHAPTER 4

THE EFFECTS OF CONTEXT ON DECEPTIVE OPINION SPAM

4.1 Introduction

Past deception research has narrowed in on a number of possible linguistic cues to deception, which we explored in Chapter 3, including: (a) *decreased spatial detail*, consistent with theories of reality monitoring [51]; (b) *increased negative emotion terms*, possibly due to "leakage" of guilt and other negative emotions associated with lying [28, 39]; and (c) *decreased first-person singular pronoun use*, often attributed to psychological distancing [39, 78]. However, recent research has found that these cues are not universally observed across settings, or **contexts**, leading some researchers to argue that cues to deception are context-dependent [7, 78, 124].

Indeed, we found in Chapter 3 that Machine Learning classifiers trained on the words of each review, or content *n*-gram features, rely heavily on context-specific features in the training data. For example, our classifiers assigned large weights to the features `hilton` and `chicago`, when trained on reviews of the HILTON CHICAGO hotel. Accordingly, it is unclear whether our classifiers will perform well at detecting deception in contexts that are different from the context of the training data.

In this chapter, we investigate the generalizability of our Machine Learning classifiers, introduced in Chapter 3, by measuring the influence that the context of a deception has on classification performance. Specifically, we consider deceptive opinion spam across three contextual dimensions:

1. **Sentiment**: Do linguistic cues to deceptive opinion spam vary with the *sentiment* of an opinion, and if so, what influence does the sentiment of an

opinion have on Machine Learning classification performance?

2. **Domain**: Do linguistic cues to deceptive opinion spam vary with the *domain*, or topic, of a review, and if so, how important is it that Machine Learning classifiers are trained and evaluated on data of the same domain?

3. **Domain Expertise**: Do authors with expert-level domain knowledge, or *domain experts*, produce deceptive opinion spam with language that is different from that of non-experts, and if so, how can we detect deceptive opinion spam written by domain experts?

First, in Section 4.2, we consider differences between positive- and negative-sentiment deceptive opinion spam. We find that Machine Learning classification performance suffers when classifiers are trained and tested on reviews of opposite sentiments, while classifiers trained on reviews of both sentiments, jointly, perform best overall. We then compare language features across positive- and negative-sentiment reviews, and confirm that some cues to deception are dependent on the sentiment; for example, DECEPTIVE reviews generally *exaggerate* the underlying sentiment of the review.

In Section 4.3, we explore the effects of domain-change on classifier performance, using the classifiers learned in Chapter 3 and the restaurant review data introduced in Sections 2.3.3 and 2.5.2. We find that classifiers trained and tested on restaurant reviews, in general, outperform cross-domain classifiers that are trained on hotel reviews and tested on restaurant reviews. However, we also find that out-of-domain (HOTEL) data can be leveraged to improve classification performance, when in-domain (RESTAURANT) training data is limited.

Finally, in Section 4.4, we investigate how deceptive opinion spam is influenced by the domain expertise of the reviewer. We compare three kinds of reviews: (a) DECEPTIVE reviews written by crowdsourced Mechanical Turk

workers (*non-experts*); (b) DECEPTIVE reviews written by hotel employess (*domain experts*), who have expert-level knowledge of their own hotels, in particular, and the hotels domain, in general; and (c) TRUTHFUL reviews written by TripAdvisor users. We find that the three kinds of reviews are distinct in their language use, and that linguistic cues to deceptive opinion spam written by hotel employees (domain experts) and crowdsourced workers (non-experts) differ. Machine Learning classifiers, however, can distinguish between the three classes with nearly 70% accuracy, or between any two out of three classes with close to 80% accuracy.

4.2 Sentiment and Deception

In this section, we explore the relationship between sentiment and deception. We perform our analysis on the *positive*-sentiment reviews used in Chapter 3, combined with the *negative*-sentiment reviews introduced in Sections 2.3.2 (DECEPTIVE) and 2.5.1 (TRUTHFUL).

First, we evaluate human deception detection performance on the *negative*-sentiment reviews, and find that humans are poor judges of deception, in line with our findings for the *positive*-sentiment reviews in Section 3.3. We then evaluate Machine Learning approaches to detecting negative deceptive opinion spam, and compare classifiers trained and tested on reviews of the same sentiment, to those trained and tested on reviews of opposite sentiments. Finally, we explore the interaction between sentiment and three hypothesized linguistic features of deception, discussed in Section 4.1: (a) decreased spatial detail; (b) increased negative emotion terms; and (c) changes in first-person singular use.

Table 4.1: Deception detection performance, incl. (P)recision, (R)ecall, and $(F)_1$-score, for three human judges and two meta-judges on 160 *negative*-sentiment reviews. Boldface denotes the largest value in each column. MAJORITY and SKEPTIC meta-judges are described in Section 3.3.1.

		TRUTHFUL			DECEPTIVE		
	Accuracy	**P**	**R**	**F**	**P**	**R**	**F**
JUDGE 1	65.0	65.0	65.0	65.0	65.0	65.0	65.0
JUDGE 2	61.9	63.0	57.5	60.1	60.9	66.3	63.5
JUDGE 3	57.5	57.3	58.8	58.0	57.7	56.3	57.0
MAJORITY	**69.4**	70.1	**67.5**	**68.8**	**68.7**	71.3	**69.9**
SKEPTIC	58.1	**78.3**	22.5	35.0	54.7	**93.8**	69.1

4.2.1 Human Performance

We first evaluate human deception detection performance on our *negative* deceptive opinion spam dataset, introduced in Section 2.3.2 (DECEPTIVE) and Section 2.5.1 (TRUTHFUL). Following our approach from Section 3.3 for *positive*-sentiment reviews, we ask three volunteer undergraduate university students to read and make assessments on a subset of the data, containing all 20 DECEPTIVE and 20 TRUTHFUL negative-sentiment reviews from each of four hotels (160 reviews total).

Results and Discussion

Performance for the three human judges is given in Table 4.1. Similarly to the positive-sentiment reviews, we find that human judges perform poorly at identifying negative deceptive opinion spam. Nevertheless, only JUDGE 3 fails to reject the null hypothesis of performing at-chance (two-tailed binomial $p = 0.07$). Still, while the best human judge is accurate 65% of the time, inter-annotator agreement between the judges is only *slight* at 0.07, computed using Fleiss' kappa [59]. Moreover, the highest pairwise inter-annotator agreement is only

0.26, computed using Cohen's kappa, between JUDGE 1 and JUDGE 2. These low agreements suggest that even when human judges perform better than chance at detecting deceptive opinion spam, they are annotating different reviews as deceptive, i.e., few reviews are consistently identified as deceptive.

4.2.2 Machine Learning Classifier Performance

We found in Chapter 3 that Machine Learning classifiers can detect *positive* deceptive opinion spam with state-of-the-art performance. However, we also found that sentiment features were important in those models (see Section 3.6.3), and it is therefore unclear how well models trained on reviews of one sentiment will generalize to reviews of the opposite sentiment. In this subsection, we evaluate the performance of Machine Learning classifiers in three settings, depending on whether we train and test classifiers on: (a) reviews of opposite sentiments (OPPOSITE-SENT); (b) reviews of the same sentiment (SAME-SENT); or (c) reviews of both sentiments, jointly (JOINT-SENT).

Experimental Setup

In the OPPOSITE-SENT setting, each classifier is trained on all 800 reviews of one sentiment, and tested on the held out set of 800 reviews of the opposite sentiment. In the SAME-SENT setting, we employ a 5-fold cross-validation (CV) procedure, described in Section 3.6.1, to evaluate classification performance when train and test sentiments are the same. Finally, we consider two JOINT-SENT settings, in which we train classifiers on both positive- and negative-sentiment reviews, but with varying training set sizes. In the first JOINT-SENT setting, we train classifiers on 400 reviews of each sentiment (800 reviews total), and test them on the held out set of 800 reviews. In the second JOINT-SENT setting, we

employ the 5-fold CV procedure, as before, but on the full combined set of 1600 reviews (800 positive-sentiment and 800 negative-sentiment).

We train and evaluate linear Support Vector Machine (SVM) classifiers with unigram and bigram features (BIGRAMS$^+$), which outperformed most other feature sets in the positive-sentiment setting (see Section 3.6.2). We tune all SVM cost parameters, C, by nested CV on the training data. Note that the classification performance reported previously in Table 3.2 is slightly different from the positive-sentiment SAME-SENT results reported here, due to an updated tokenizer used in these experiments.

Results and Discussion

Results appear in Table 4.2. The results confirm that n-gram–based SVM classifiers can detect *negative* deceptive opinion spam in a balanced dataset with performance far surpassing that of untrained human judges (*cf.* Table 4.1). Furthermore, a one-tailed sign test shows that classifiers trained and tested on reviews of different sentiments (OPPOSITE-SENT) perform significantly worse ($p = 0.013, 0.001$ for POSITIVE and NEGATIVE test sentiments, respectively) than classifiers trained and tested on reviews of the same sentiment (SAME-SENT), despite having access to a greater number of training reviews.[1] This suggests that cues to deception differ depending on the sentiment of the text, which we discuss further in Section 4.2.3.

We also observe that classifiers in the JOINT-SENT setting outperform their corresponding classifiers in the OPPOSITE-SENT setting, even when the number of reviews in the training set is controlled (800 reviews). Moreover, training classifiers on the full 1600-review JOINT-SENT dataset results in performance

[1] SAME-SENT classifiers are trained on 80% of the available reviews (4 CV folds), whereas OPPOSITE-SENT classifiers are trained on 100% of the available reviews.

Table 4.2: SVM classifier performance for models trained and tested on reviews of the same sentiment, different sentiments, or both (joint) sentiments. Evaluation is performed via cross-validation or on held out reviews, depending on the setting.

		Number and Type of Train Reviews			TRUTHFUL			DECEPTIVE		
Test Sentiment	**Setting**	POSITIVE	NEGATIVE	**Accuracy**	**P**	**R**	**F**	**P**	**R**	**F**
POSITIVE (800 reviews)	SAME-SENT	800	-	89.5	90.3	88.5	89.4	88.7	90.5	89.6
	OPPOSITE-SENT	-	800	81.4	76.3	91.0	83.0	88.9	71.8	79.4
	JOINT-SENT	400	400	85.2	84.4	86.5	85.4	86.2	84.0	85.1
	JOINT-SENT	800	800	88.5	87.9	89.2	88.6	89.1	87.8	88.4
NEGATIVE (800 reviews)	SAME-SENT	-	800	86.6	86.2	87.2	86.7	87.1	86.0	86.5
	OPPOSITE-SENT	800	-	75.4	68.9	92.5	79.0	88.6	58.2	70.3
	JOINT-SENT	400	400	86.0	87.1	84.5	85.8	85.0	87.5	86.2
	JOINT-SENT	800	800	87.0	86.1	88.2	87.2	87.9	85.8	86.8

that is comparable to classifiers in the SAME-SENT setting, both when testing on positive-sentiment reviews (88.5% vs. 89.5% accuracy, two-tailed sign test $p = 0.80$) and when testing on negative-sentiment reviews (87.0% vs. 86.6% accuracy, $p = 0.97$). This result is explained in part, however, by the increased training set size (1,280 vs. 640 reviews for 4 training folds).

Unfortunately, the quantity of features in n-gram classifiers prohibits a thorough analysis of the improved generalizability of the JOINT-SENT classifiers over the sentiment-specific classifiers. However, in the next subsection (Section 4.2.3), we explore the linguistic cues to deception across sentiments, and find that some cues to deception apply across sentiments, e.g., DECEPTIVE reviews generally lack spatial details, compared to TRUTHFUL reviews. Thus, it is likely that the JOINT-SENT classifiers place greater emphasis on these sentiment-independent features, and less emphasis on sentiment-specific features.

4.2.3 Linguistic Cues to Deception Across Sentiments

We now explore how language features compare across positive- and negative-sentiment, DECEPTIVE and TRUTHFUL reviews. In Table 4.3, we consider differences in three categories of language features, by comparing the sentiment- and class-specific means and standard deviations of four LIWC categories: (a) spatial details (`space`); (b) sentiment (`pos emo` and `neg emo`); and (c) first-person singular pronouns (`i`).

Spatial Details We found in Section 3.6.3 that positive DECEPTIVE reviews used fewer spatial terms (e.g., `small`, `bathroom`, `on`, `floor`, `location`), compared to positive TRUTHFUL reviews (one-tailed t-test $p = 5 \times 10^{-5}$), because Mechanical Turk authors never experienced the hotel and had less spatial de-

Table 4.3: (M)eans and (S)tandard (D)eviations of four LIWC categories across 400 positive- and 400 negative-sentiment hotel reviews. The reported statistics correspond to percentages of total word use in any given review.

	Positive Sentiment				Negative Sentiment			
	TRUTHFUL		DECEPTIVE		TRUTHFUL		DECEPTIVE	
Category	**M (%)**	**SD (%)**	**M**	**SD**	**M**	**SD**	**M**	**SD**
space	8.07	2.63	7.36	2.51	7.71	2.59	7.44	2.11
pos emo	5.83	2.55	6.44	2.51	2.51	1.70	2.24	1.37
neg emo	0.46	0.64	0.24	0.50	1.42	1.03	1.68	1.23
i	2.18	2.04	4.36	2.96	2.85	2.23	4.47	2.83

tails available for their review [51]. This was similarly the case for the negative reviews, with more spatial details (the LIWC space category) in negative TRUTHFUL reviews, compared to negative DECEPTIVE reviews ($p = 0.049$).

Sentiment Previous research has found that deception is often correlated with increased negative emotion terms [78], possibly due to guilt on the part of the deceiver [39], and corresponding "leakage cues" [28]. In agreement with these previous findings, we find that negative DECEPTIVE reviews use more *negative* emotion terms (the LIWC neg emo cateogry), compared to negative TRUTHFUL reviews (one-tailed t-test $p = 5 \times 10^{-4}$). In contrast, we found in Chapter 3 that positive DECEPTIVE reviews were generally more *positive* in sentiment than positive TRUTHFUL reviews ($p = 4 \times 10^{-4}$). Combined, our results suggest that DECEPTIVE reviews *exaggerate* the underlying review sentiment.

First-Person Singular Pronouns We found in Section 3.6.3 that among positive-sentiment reviews, first-person singular use was more frequent in DECEPTIVE, compared to TRUTHFUL reviews, in contrast to previous studies of deception [78]. While we observe a similar effect among negative-sentiment

reviews, the magnitude of the difference is smaller in the negative reviews, compared to the positive reviews. In particular, the rates among positive and negative DECEPTIVE reviews remain similar (M=4.36% vs. M=4.47%, respectively; two-tailed t-test $p = 0.59$), but the rates increase 31% from positive TRUTHFUL to negative TRUTHFUL reviews (M=2.18% vs. M=2.85%, respectively; one-tailed t-test $p = 5 \times 10^{-6}$). One possible explanation for the increased focus on self in negative TRUTHFUL, compared to positive TRUTHFUL reviews, is that negative TRUTHFUL reviews reflect the frustration and anxiety of a real-life negative customer experience [42], which may result in increased usage of first-person singular [97].

4.3 Domain Topic and Deception

It is also important when training Machine Learning classifiers to consider the domain (or topic) of the training data, in relation to the data on which the classifier will be evaluated. For example, we observed in Section 3.6.3 that *n*-gram–based Machine Learning classifiers rely on features that are highly specific to the domain of the training data (e.g, `hilton`, `chicago`, `hotel`). Accordingly, those classifiers are unlikely to perform well at detecting deception in domains that are very different from the training domain.

In this section, we explore how the domain of a deception influences classification performance, by training Machine Learning classifiers on reviews from one domain (HOTELS), and evaluating their ability to distinguish between DECEPTIVE and TRUTHFUL reviews from another domain (RESTAURANTS). We have chosen to use restaurants as our second domain, because of the strong similarities between restaurant and hotel reviews, which allow us to explore the subtle effects of domain change.

In general, we find that classification performance suffers when train and test domains differ. However, we also find that we can leverage out-of-domain training data, in combination with in-domain training data, to learn models that perform slightly better than those that use only in-domain data.

4.3.1 Experimental Setup

We frame our problem as a *domain adaptation* task [89], where we are given a large set of labeled training data from one domain (HOTELS), and our goal it to learn a classifier that performs well on data from a different domain (RESTAURANTS). In particular, we train our classifiers on positive-sentiment hotel reviews, used in Chapter 3, and evaluate classification performance on positive-sentiment restaurant reviews, which we introduced in Section 2.3.3 (DECEPTIVE) and Section 2.5.2 (TRUTHFUL).

We compare linear Support Vector Machine (SVM) classifiers trained on three feature sets: POS, LIWC, and BIGRAMS$^+$ (described in Section 3.5). We evaluate these classifiers on 400 restaurant reviews using the 5-fold nested cross-validation procedure from Section 3.6.1. Each test fold contains 20 DECEPTIVE and 20 TRUTHFUL reviews from each of two restaurants (80 reviews total).

We explore three training settings. In the first setting (IN-DOMAIN), we evaluate in-domain cross-validation performance on the 400 restaurant reviews. In the second setting (CROSS-DOMAIN), we train a classifier on 800 hotel reviews, i.e., out-of-domain data, and evaluate its performance on 400 held out restaurant reviews. In the third and fourth settings (JOINT-DOMAIN), we train classifiers on a combination of in-domain (RESTAURANT) and out-of-domain (HOTEL) reviews, in varying quantities, and evaluate their performance on the restaurant reviews via cross-validation.

Table 4.4: SVM classifier performance on 400 RESTAURANT reviews. Classifiers are trained on varying quantities of in-domain (RESTAURANT) and out-of-domain (HOTEL) reviews.

	Number and Type of Train Reviews				TRUTHFUL			DECEPTIVE		
Setting	HOTELS	RESTAURANTS	**Features**	**Accuracy**	**P**	**R**	**F**	**P**	**R**	**F**
IN-DOMAIN	-	400	POS	72.8	72.0	74.5	73.2	73.6	71.0	72.3
			LIWC	76.2	77.5	74.0	75.7	75.1	78.5	76.8
			BIGRAMS$^+$	84.8	82.9	87.5	85.2	86.8	82.0	84.3
CROSS-DOMAIN	800	-	POS	70.8	66.8	82.5	73.8	77.1	59.0	66.9
			LIWC	75.2	73.9	78.0	75.9	76.7	72.5	74.5
			BIGRAMS$^+$	81.5	79.7	84.5	82.0	83.5	78.5	80.9
JOINT-DOMAIN	800	200	POS	72.2	68.5	82.5	74.8	78.0	62.0	69.1
			LIWC	77.2	77.1	77.5	77.3	77.4	77.0	77.2
			BIGRAMS$^+$	85.2	83.4	**88.0**	85.6	**87.3**	82.5	84.8
JOINT-DOMAIN	800	400	POS	72.2	69.3	80.0	74.2	76.3	64.5	69.9
			LIWC	78.2	78.4	78.0	78.2	78.1	78.5	78.3
			BIGRAMS$^+$	**86.8**	**86.9**	86.5	**86.7**	86.6	**87.0**	**86.8**

4.3.2 Results and Discussion

Results are given in Table 4.4. Reference performance is given by our IN-DOMAIN setting, corresponding to cross-validation on the 400 restaurant reviews. In this setting, we once again find that BIGRAM$^+$ features perform best, similar to our findings for hotel reviews in Section 3.6.

Cross-domain results are given by our CROSS-DOMAIN setting, where a classifier is trained on 800 out-of-domain (HOTEL) reviews and tested on 400 held out in-domain (RESTAURANT) reviews. We observe that classification performance is worse in the CROSS-DOMAIN, compared to the IN-DOMAIN training setting, despite using double the training data (800 out-of-domain vs. 400 in-domain reviews). Notably, while BIGRAM$^+$ features still perform best, POS and LIWC features are more robust across domains, i.e., there is less difference in performance between in-domain and out-of-domain training settings for these features. This confirms that POS and LIWC features are more domain independent than *n*-gram features, although they remain uncompetitive, in general.

In our third and fourth settings (JOINT-DOMAIN), we evaluate classifiers trained on a combination of in-domain (RESTAURANT) and out-of-domain (HOTEL) reviews. We find that a joint-domain classifier trained on just 200 in-domain (RESTAURANT) reviews, combined with 800 out-of-domain (HOTEL) reviews, achieves higher accuracy than using a fully in-domain classifier trained on the full 400 in-domain (RESTAURANT) reviews (85.2% vs. 84.8% accuracy, respectively, using BIGRAM$^+$ features), although the increase is not significant (one-tailed sign test $p = 0.36$). Moreover, combining the full 400 in-domain (RESTAURANT) and 800 out-of-domain (HOTEL) reviews results in even better performance (86.2% vs. 84.8% accuracy, using BIGRAM$^+$ features), although the increase is not significant (one-tailed $p = 0.29$).

Table 4.5: (M)eans and (S)tandard (D)eviations of six selected LIWC categories across 800 positive HOTEL reviews and 400 positive RESTAURANT reviews. The reported statistics correspond to percentages of total word use in any given review.

	HOTEL Reviews				RESTAURANT Reviews			
	TRUTHFUL		DECEPTIVE		TRUTHFUL		DECEPTIVE	
Category	**M (%)**	**SD (%)**	**M**	**SD**	**M**	**SD**	**M**	**SD**
home	2.88	1.57	2.60	1.70	0.42	0.63	0.63	0.78
ingest	1.30	1.40	1.08	1.25	3.94	2.02	3.99	1.93
space	8.07	2.63	7.36	2.51	5.77	2.41	4.67	2.18
pos emo	5.83	2.55	6.44	2.51	5.13	2.26	6.48	2.34
neg emo	0.46	0.64	0.24	0.50	0.46	0.61	0.40	0.62
i	2.18	2.04	4.36	2.96	2.03	1.87	4.42	2.21

4.3.3 Linguistic Cues to Deception Across Domains

We now explore how language features compare across hotel and restaurant reviews. In Table 4.5, we consider differences in four categories of language features, by comparing the domain- and class-specific means and standard deviations of six LIWC categories: (a) domain-specific details (home and ingest); (b) spatial details (space); (c) sentiment (pos emo and neg emo); and (d) first-person singular pronouns (i).

Domain-Specific Details First, we observe some domain-specific differences between hotel and restaurant reviews. Hotel reviews, for example, compared to restaurant reviews, use significantly more terms relating to *personal concerns*, such as the LIWC home category (M=2.88% vs. M=0.42% for TRUTHFUL reviews and M=2.60% vs. M=0.63% for DECEPTIVE reviews, respectively). On the other hand, restaurant reviews, compared to hotel reviews, generally use more terms relating to *biological processes*, such as the LIWC ingest category (M=3.94%

vs. M=1.30% for TRUTHFUL reviews and M=3.99% vs. M=1.08% for DECEPTIVE reviews, respectively). However, there is little difference between DECEPTIVE and TRUTHFUL reviews within these categories.

Spatial Details We found in Section 3.6.3 and Section 4.2.3 that DECEPTIVE reviews, compared to TRUTHFUL reviews, use fewer spatial terms (e.g., `small`, `floor`, `location`, and the LIWC `space` category). Similarly, we find that DECEPTIVE restaurant reviews, compared to TRUTHFUL restaurant reviews, use significantly fewer spatial terms (M=4.67% vs. M=5.77%, respectively, for the LIWC `space` category; one-tailed t-test $p = 1.4 \times 10^{-6}$). However, we also observe that spatial terms are very significantly less frequent in restaurant reviews, on average, compared to hotel reviews (M=4.67% vs. M=7.36%, respectively, for DECEPTIVE reviews, and M=5.77% vs. M=8.07%, respectively, for TRUTHFUL reviews).

Sentiment We observe that DECEPTIVE restaurant reviews are more positive in sentiment, on average, compared to TRUTHFUL restaurant reviews (M=6.48% vs. M=5.13%, respectively; one-tailed t-test $p = 5.0 \times 10^{-9}$). This finding is similar to our findings for positive-sentiment hotel reviews, where DECEPTIVE reviews *exaggerate* the underlying sentiment of the review (see Section 3.6.3). Notably, hotel and restaurant reviews use sentiment terms with similar relative frequencies.

First-Person Singular Pronouns For restaurant reviews, we once again observe that DECEPTIVE reviews use first-person singular pronouns with greater frequency, relative to TRUTHFUL reviews (M=4.42% vs. M=2.03%, respectively; one-tailed t-test $p = 2.2 \times 10^{-16}$). However, the relative difference in first-person

singular pronoun use between hotel and restaurant reviews is small.

4.4 Domain Expertise and Deception

The final contextual dimension that we explore is the effect of domain expertise on deception. Specifically, this section considers deceptive reviews written by *hotel employees*, who are domain experts in the HOTELS domain, and who possess substantial spatial knowledge of their own hotels. Importantly, hotel employees are a plausible source of real-world fake hotel reviews, and the domain knowledge possessed by these employees may enable them to craft deceptive reviews that are more difficult for classifiers to detect, compared to the crowdsourced DECEPTIVE reviews considered thus far.

For example, we found in Chapter 3 and Section 4.2 that crowdsourced DECEPTIVE hotel reviews, compared to TRUTHFUL hotel reviews, include fewer spatial details (e.g., `small`, `bathroom`, `on`, `floor`, `location`); accordingly, classifiers trained on these reviews weight spatial details heavily in the direction of the TRUTHFUL class. It is not clear, however, whether a lack of spatial detail indicates deception, in general, or if it is instead an artifact of reviews written by crowdsourced workers, who have no experience with the hotel they are reviewing. In this section, we investigate some of the similarities and differences between domain expert deceptive opinion spam, crowdsourced deceptive opinion spam, and TRUTHFUL reviews.

4.4.1 Experimental Setup

In previous sections and chapters, we have considered a binary classification task, in which the goal is to predict whether a given review is DECEPTIVE (de-

ceptive opinion spam) or TRUTHFUL (not deceptive opinion spam). In this section, we reframe our task as a **multi-class classification task**, in which the goal is to classify reviews according to their source: TripAdvisor (TRUTHFUL), Mechanical Turk (MTURK), or a hotel employee (EMPLOYEE).

We use a One-versus-Rest (OvR) classification scheme, in which we train m binary classifiers, such that each classifier, f_i, for $i \in [1, m]$, is trained to distinguish between class i, on the one hand, and all classes except i, on the other. To make an m-way classification decision, we then choose the class, c, corresponding to the classifier, f_c, with the most confident prediction. OvR approaches have been shown to produce state-of-the-art performance, compared to other multi-class approaches, such as Multinomial Naïve Bayes [106] and One-vs-One classification schemes [107]. We train OvR multi-class linear SVM classifiers on three feature sets: POS, LIWC, and BIGRAMS$^+$ (described in Section 3.5).

We use the dataset that we introduced in Section 2.4. In particular, our dataset contains EMPLOYEE reviews written by two employees from each of seven hotels. Accordingly, we report 7-fold cross-validation performance, where each fold contains the EMPLOYEE reviews written by two employees from a single hotel, as well as the corresponding TRUTHFUL and MTURK reviews.

4.4.2 Results and Discussion

Multi-class classification results are given in Table 4.6. We report both OvR (3-CLASS) performance, and the performance of three One-versus-One binary classifiers, trained to distinguish between each pair of classes.

Once again, BIGRAMS$^+$ feature perform best, approaching 70% accuracy on the 3-class classification task, where a random-guess baseline is just 33.3% accurate. We also observe that each of the three One-versus-One binary clas-

Table 4.6: Multi-class SVM classifier performance on 280 mixed-sentiment reviews from each of three sources: TripAdvisor (TRUTHFUL), Mechanical Turk (MTURK), and hotel employees (EMPLOYEE).

			TRUTHFUL			MTURK			EMPLOYEE		
Setting	**Features**	**Accuracy**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
3-CLASS	POS	49.2	52.7	64.6	58.0	48.1	45.7	46.9	45.3	37.5	41.0
	LIWC	57.2	61.1	61.3	61.2	53.0	59.6	56.1	58.2	50.7	54.2
	BIGRAMS$^+$	69.9	69.7	73.7	71.6	69.2	71.4	70.3	71.0	64.6	67.7
TRUTHFUL vs. MTURK	BIGRAMS$^+$	80.1	79.9	79.9	79.9	80.4	80.4	80.4	–	–	–
TRUTHFUL vs. EMPLOYEE	BIGRAMS$^+$	80.0	77.1	84.7	80.7	–	–	–	83.4	75.4	79.2
MTURK vs. EMPLOYEE	BIGRAMS$^+$	77.5	–	–	–	74.1	84.6	79.0	82.1	70.4	75.8

Table 4.7: (M)eans and (S)tandard (D)eviations of four selected LIWC categories across 140 positive- and 140 negative-sentiment hotel reviews from three sources: TripAdvisor (TRUTHFUL), Mechanical Turk (MTURK), and hotel employees (EMPLOYEE). The reported statistics correspond to percentages of total word use in any given review.

		TRUTHFUL		MTURK		EMPLOYEE	
Sentiment	**Category**	**M (%)**	**SD (%)**	**M**	**SD**	**M**	**SD**
POSITIVE	space	7.48	2.92	7.22	2.38	7.60	3.48
	posemo	5.87	1.95	6.05	2.29	6.54	3.99
	negemo	0.44	0.69	0.29	0.48	0.27	0.67
	i	2.44	2.36	3.61	2.64	2.89	3.03
NEGATIVE	space	7.34	2.53	7.25	2.54	6.81	2.99
	posemo	3.58	1.91	2.14	1.80	2.65	2.04
	negemo	1.15	1.16	1.68	1.10	1.54	1.53
	i	2.60	2.25	4.23	3.16	4.19	3.48

sifiers perform significantly better than chance, suggesting that TRUTHFUL, MTURK and EMPLOYEE reviews are, in fact, three distinct classes. In particular, BIGRAMS$^+$ features are 77.5% accurate at distinguishing between MTURK and EMPLOYEE reviews, despite that both kinds of reviews are deceptive opinion spam, and that the two datasets are controlled to have equal numbers of reviews across sentiments and hotels. We explore the differences and similarities between MTURK and EMPLOYEE reviews further in Section 4.4.3.

4.4.3 Linguistic Cues to Deception by Domain Experts

We now explore how language features compare across TRUTHFUL, MTURK and EMPLOYEE reviews. In Table 4.7, we consider differences in three categories of language features, by comparing the class-specific means and standard deviations of four LIWC categories: (a) spatial details (space); (b) sentiment (pos emo and neg emo); and (c) first-person singular pronouns (i).

We additionally separate the results by the sentiment of the review. This is important, because our hotel employee authors wrote positive-sentiment reviews when reviewing their own hotels, and negative-sentiment reviews when reviewing their competitor's hotels (see the description of the dataset in Section 2.4). Thus, we expect positive-sentiment EMPLOYEE reviews to exhibit greater spatial awareness, due to reviewing their own hotel, compared to negative-sentiment EMPLOYEE reviews.

Spatial Details In agreement with our expectations, hotel employees encode significantly more spatial information into their lies when reviewing their own hotel (i.e., the positive-sentiment EMPLOYEE reviews; M=7.60%), compared to when reviewing a competitor's hotel (i.e., the negative-sentiment EMPLOYEE reviews; M=6.81%; one-tailed t-test $p = 0.02$). These findings suggest that a lack of spatial details may not be a universal cue to deception, but rather, it is an artifact of reviews written by authors who have no prior experience with the hotel they are reviewing. Moreover, it appears that general domain expertise does not compensate for this lack of prior experience, as demonstrated by the lack of spatial details in the negative-sentiment EMPLOYEE reviews.

Also in line with our expectations, and our previous findings in Sections 3.6.3, 4.2.3 and 4.3.3, we observe that MTURK reviews have fewer spatial terms, on average, compared to TRUTHFUL reviews, although the differences are not statistically significant in this case, potentially due to the smaller dataset size (one-tailed t-test $p = 0.20, 0.39$, for positive- and negative-sentiment reviews, respectively).

Sentiment With respect to sentiment, we once again observe that DECEPTIVE reviews (both MTURK and EMPLOYEE), relative to TRUTHFUL reviews, exag-

gerate the underlying sentiment of the review. All pairwise differences in Table 4.7 between DECEPTIVE (MTURK or EMPLOYEE) and TRUTHFUL reviews, for the `posemo` and `negemo` categories, are significant at the $p < 0.05$ level, except for the difference in `posemo` between positive-sentiment MTURK reviews (M=6.05%) and positive-sentiment TRUTHFUL reviews (M=5.87%), where the difference is not significant ($p = 0.25$).

First-Person Singular Pronouns We also observe increased use of first-person singular pronouns in both of the DECEPTIVE review conditions (MTURK and EMPLOYEE), relative to TRUTHFUL reviews. In particular, all pairwise differences in Table 4.7 between DECEPTIVE (MTURK or EMPLOYEE) and TRUTHFUL reviews, for the `i` category, are significant at a $p < 0.05$ level, except between positive-sentiment EMPLOYEE and TRUTHFUL reviews, where the difference is not significant (M=2.89% vs. M=2.44%, respectively; $p = 0.09$). Thus, increased usage of first-person singular pronouns appears to be a relatively consistent cue to deceptive opinion spam in our data.

4.5 Discussion and Chapter Summary

In this chapter, we have demonstrated how three contextual factors, including the *sentiment* of a review, the *domain* of a review, and the *domain expertise* of the reviewer, can influence cues to deception, and the ability of Machine Learning classifiers to detect deceptive opinion spam.

First, we have shown that DECEPTIVE reviews vary across sentiments, and generally *exaggerate* the underlying review sentiment, compared to TRUTHFUL reviews. We have also shown that classifiers trained and tested on reviews of the same sentiment outperform classifiers trained and tested on reviews of opposite

sentiments. Notably, classifiers trained on reviews of both sentiments, jointly, perform best overall.

Next, we have shown that reviews of different domains vary in their use of certain categories of words; for example, restaurant reviews include fewer spatial details than hotel reviews, but more words relating to biological processes, such as the LIWC `ingest` category. Nevertheless, classifiers trained on reviews from one domain (e.g., HOTELS) perform reasonably well at detecting deceptive reviews in another domain (e.g., RESTAURANTS). Moreover, we show that in cases where in-domain training data is limited, we can leverage out-of-domain training data to improve classification performance.

Lastly, we have shown that reviews written by hotel employees (domain experts) are distinct from both TRUTHFUL and crowdsourced (i.e., non-expert) DECEPTIVE reviews. We have shown that a One-versus-Rest multi-class classifier can distinguish between the three classes with nearly 70% accuracy, while binary classifiers can distinguish between any two out of three classes with close to 80% accuracy. Finally, we have shown that a lack of spatial details may be an artifact of reviews written by authors who have no experience with the hotel they are reviewing, rather than a universal cue to deceptive opinion spam.

In general, our findings in this chapter highlight the importance of modeling the context of a lie, in studies of deception, and provide examples of some of the ways in which cues to deception may vary across contexts.

CHAPTER 5

ESTIMATING THE PREVALENCE OF DECEPTIVE OPINION SPAM

5.1 Introduction

Perhaps surprisingly, relatively little is known about the actual *prevalence*, or rate, of deception in online review communities, and less still is known about the factors that can influence it. On the one hand, the relative ease of producing reviews, combined with the pressure for businesses, products, and services to be perceived in a positive light, suggests that deceptive reviews may be quite common. On the other hand, consumers increasingly rely on online reviews to make purchase decisions [18, 46], suggesting that consumers find the information contained in those reviews to be accurate.

Estimating the prevalence of deception in online review communities is challenging, however, because gold standard annotations of deception are not available for real-world reviews, and neither human judgements nor self-reports of deception are reliable either (see the discussion in Chapter 2). In this chapter, we propose an approach for estimating the prevalence of deception in an online review community, based on the output of the Machine Learning classifiers in Chapter 3.

More generally, we investigate the prevalence of deceptive opinion spam in online review communities through three questions:

1. **Prevalence Estimation**: In the absence of gold standard annotations, can we use a noisy deception classifier to estimate the prevalence of deceptive opinion spam in a review community?
2. **Influencing Factors**: What factors influence the prevalence of deception in a review community?

3. **Prevalence Reduction**: What actions, if any, can be taken to reduce the prevalence of deception in a review community?

First, we introduce the *Bayesian Prevalence Model* (see Section 5.3)—a generative model of deception that jointly models the uncertainty of a deception classifier, as well as the ground-truth deceptiveness of each review. Inference for this model, which we perform via Gibbs sampling, allows us to estimate bounds on the prevalence of deception in a review community, without requiring gold standard annotations of deception.

We then reason about the factors that influence deception prevalence in online review communities (see Section 5.4). We argue that reviews are signals to the true, unknown quality of a product or service offering, and act to diminish the information asymmetry between past and prospective customers [44, 117]. Accordingly, deceptive opinion spam is a *false signal*, and the prevalence of deception among online reviews is a function of the *signaling costs* and *benefits* associated with posting fake reviews. We hypothesize that review communities with low signaling cost, such as communities that make it easy to post a review, and large benefits, such as highly trafficked sites, will attract more deceptive opinion spam than will communities with higher signaling costs, such as communities that establish additional requirements for posting reviews, and lower benefits, such as low site traffic.

We apply our approach to positive-sentiment Chicago hotel reviews in six online hotel review communities: Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor, and Yelp. We find first that the prevalence of deception indeed varies by community. However, because it is not possible to validate these estimates empirically (i.e., gold standard rates of deception in each community are unknown), we focus our discussion instead on the relative differences in the es-

timated rates of deception between communities. Here, the results confirm our hypotheses and suggest that deception is most prevalent in communities with a low signal cost. Importantly, when measures are taken to increase a community's signal cost, we find reductions in our estimated rates of deception in that community.

5.2 Related Work

The focus of most previous spam research in the context of online reviews has been on *detection* (see related work in Section 3.2). Indeed, there are few empirical, scholarly studies of the prevalence of deceptive opinion spam.

Notably, as we discussed in Section 2.2.2, Mayzlin et al. [67] reason about the economic incentives for posting deceptive hotel reviews by comparing a hotel's rating distribution on TripAdvisor, where anyone can create an account and post a review, to the same hotels' rating distribution on Expedia, where a verified purchase is required before a user can post a review. They apply a difference of differences approach to control for factors specific to each review community, and examine the relative degree of review manipulation between different categories of hotels, e.g., independent vs. chain hotels, small vs. large owner hotels, and hotels with small vs. large management companies. Importantly, they find review manipulation to be relatively insignificant, with the biggest difference between hotel categories occurring between small independent hotels and large chain hotels, with the former posting an estimated 7 additional fake positive reviews (out of 120), relative to the latter.

Our approach is also related to studies of disease prevalence, in which gold standard diagnostic testing is either too expensive, or impossible to perform [53, 54]. In such cases, the prevalence of disease in a population is esti-

mated using a combination of an imperfect diagnostic test, and estimates of the test's positive and negative recall rates. For example, Joseph et al. [54] propose a Bayesian model, similar to our own, that models the total number of true positives and false negatives produced by an imperfect diagnostic test. However, while pilot experiments confirm that their model performs similarly to our own, the generative story of our model, given in Section 5.3, is comparatively more intuitive.

Finally, work in Machine Learning has explored classification settings where the distribution of the classes in the test data is different from the class distribution observed at train time. In particular, Saerens et al. [111] propose an approach to estimate the unknown class distribution in a test set, based on Expectation Maximization (EM), using a classifier's output probabilities on the test data, and the known class distribution in the training data. Unfortunately, the EM approach does not produce bounds on its estimates, as our approach does, although the point estimates of the two approaches are similar.

5.3 Bayesian Prevalence Model

The Bayesian Prevalence Model (BAYES) uses the output of our Machine Learning classifiers, introduced in Chapter 3, to estimate the prevalence of deception in a group of reviews. Unfortunately, our classifiers can produce both false positive and false negative predictions, and, therefore, cannot be relied on directly. Moreover, if the probability of a false positive prediction is different from the probability of a false negative prediction, then the error of the estimates produced by our classifier will vary depending on the true rate of deception.

To address these challenges, BAYES jointly models our classifier's false positive and false negative rates, as well as the true rate of deception. Specifically,

BAYES models the generative process through which deception occurs, including the true (latent) **rate of deception** (π^*), the classifier's true (latent) deceptive recall, or **sensitivity** (η^*), and truthful recall, or **specificity** (θ^*). Formally, BAYES assumes that our data was generated according to the following generative story, where for each review, indexed by i, the (latent) **ground truth label** is given by $y_i \in \{0, 1\}$, and the (observed) **classifier prediction** is given by $f(\mathbf{x}_i)$:

- Sample the true rate of deception: $\pi^* \sim \text{Beta}(\boldsymbol{\alpha})$
- Sample the classifier's true sensitivity: $\eta^* \sim \text{Beta}(\boldsymbol{\beta})$
- Sample the classifier's true specificity: $\theta^* \sim \text{Beta}(\boldsymbol{\gamma})$
- For each review, indexed by i:
  - Sample the review's ground truth deception label:

$$y_i \sim \text{Bernoulli}(\pi^*)$$

  - Sample the classifier's prediction:

$$f(\mathbf{x}_i) \sim \begin{cases} \text{Bernoulli}(\eta^*) & \text{if } y_i = 1 \\ \text{Bernoulli}(1 - \theta^*) & \text{if } y_i = 0 \end{cases}$$

The corresponding graphical model is given in plate notation in Figure 5.1, where shaded and unshaded nodes indicate observed and latent variables, respectively, and directed edges denote dependencies between variables. Notice that by placing Beta prior distributions on π^*, η^*, and θ^*, BAYES enables us to encode our prior knowledge about the true rate of deception, as well as our uncertainty about the estimates of the classifier's sensitivity and specificity. This is discussed further in Section 5.5.2.

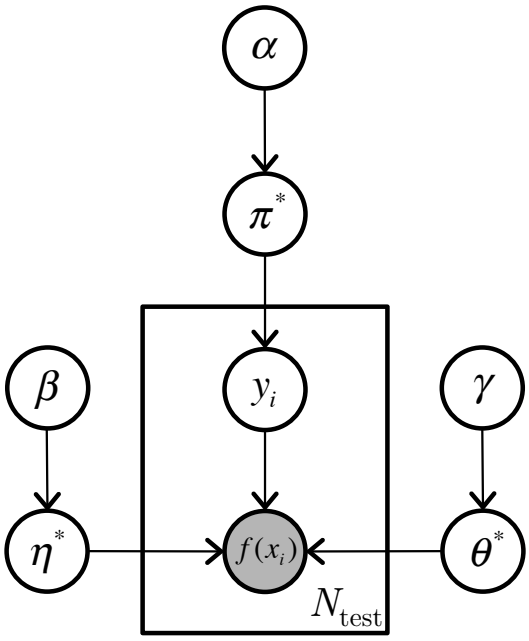


Figure 5.1: The Bayesian Prevalence Model in plate notation. Shaded nodes represent observed variables, and arrows denote dependence. For example, $f(\mathbf{x}_i)$ is observed, and depends on η^*, θ^*, and y_i.

Inference

While exact inference is intractable for the Bayesian Prevalence Model, an alternative way of approximating the desired posterior distribution is with Markov Chain Monte Carlo Gibbs sampling. Gibbs sampling works by sampling each variable, in turn, from the conditional distribution of that variable given all other variables in the model. After repeating this procedure for many iterations, the desired posterior distribution can be approximated from samples in the chain by: (1) discarding a number of initial *burn-in* iterations, and (2) thinning the number of remaining samples according to a *sampling lag*, since adjacent samples in the chain are often highly correlated.

The conditional distributions of each variable, given the others, can be derived from the joint distribution. Based on the graphical representation of BAYES, given in Figure 5.1, the joint distribution of the observed and latent vari-

ables is given by:

$$\begin{aligned}
&\Pr(f(\mathbf{x}), \mathbf{y}, \pi^*, \eta^*, \theta^*; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \\
&\quad = \Pr(f(\mathbf{x}) \mid \mathbf{y}, \eta^*, \theta^*) \cdot \Pr(\mathbf{y} \mid \pi^*) \cdot \Pr(\pi^* \mid \boldsymbol{\alpha}) \cdot \Pr(\eta^* \mid \boldsymbol{\beta}) \cdot \Pr(\theta^* \mid \boldsymbol{\gamma}), \qquad (5.1)
\end{aligned}$$

where each term is given by the sampling distributions specified in the generative story in Section 5.3.

A common technique to simplify the joint distribution, and the sampling process, is to integrate out (collapse) variables that do not need to be sampled. If we integrate out π^*, η^*, and θ^* from Equation 5.1, we can derive a collapsed Gibbs sampler that only needs to sample the y_i's at each iteration. The resulting sampling equations, and the corresponding Bayesian Prevalence Model estimate of the prevalence of deception, π^*, are given in greater detail in Appendix A.1.

5.4 Signal Theory

In terms of economic theory, the role of online reviews is to reduce the inherent *information asymmetry* between prospective customers, on the one hand, and past customers and sellers, on the other [44]. It follows that if reviews regularly failed to reduce this information asymmetry, or, worse, convey false information, then they would cease to be of value to the user. And while users do, in fact, value online reviews [18, 46], there remains widespread concern about deception in online review communities.

Unfortunately, our understanding of the prevalence of deceptive opinion spam is limited (see related work in Section 5.2). Moreover, it is not possible to empirically validate the estimates produced by our Bayesian Prevalence Model (Section 5.3), because we do not have ground truth labels of deception for real-world reviews. However, by framing reviews as *signals* to a product's

true (unknown) quality [44, 117], we can reason about the prevalence of deceptive opinion spam as a function of the costs and benefits associated with posting fake reviews.

First, consider the *benefits* associated with posting deceptive opinion spam. A business is benefited by posting a fake positive review when a prospective customer reads and is influenced by that review, especially when making a purchase decision. Moreover, this **exposure benefit** varies by review community, and is proportional to the size of the community's audience. Therefore, we argue that deception will be more prevalent in review communities with a *high exposure benefit*, such as highly trafficked sites, compared to communities with *low exposure benefit*, for example, sites with low traffic.

We next consider the *costs* associated with posting deceptive opinion spam. In general, the costs of deception are broad and may include psychological and emotional costs [23], as well as reputational costs [67]. Here, we consider only the **posting cost**, or the direct monetary and time costs associated with posting a review in a review community. Posting costs also vary by review community, with some communities *verifying* purchases before allowing reviews to be posted, and others hiding or filtering reviews written by new or inexperienced reviewers. We argue that review communities that implement these costs, i.e., communities with a *high posting cost*, will contain less deceptive opinion spam compared to communities with a *low posting cost*.

Observe that both posting costs and exposure benefits depend entirely on the review community. An overview of these factors for each of the six review communities is given in Table 5.1. Note that we have used the number of Chicago hotel reviews in each community to determine the exposure benefit.

Table 5.1: Signal costs associated with six online review communities, sorted approximately from highest signal cost to lowest. The posting cost is *High* if users are required to purchase a product before reviewing it, and *Low* otherwise. The exposure benefit is *Low*, *Medium*, or *High* based on the number of reviews in the community (see Table 2.2).

Community	Posting Cost	Exposure Benefit
Orbitz	High	Low
Priceline	High	Medium
Expedia	High	Medium
Hotels.com	High	Medium
Yelp	Low	Low
TripAdvisor	Low	High

5.4.1 Hypotheses

Based on the posting costs and exposure benefits just defined, we propose two hypotheses, which we test in Section 5.6:

- **Hypothesis 1**: Review communities that have low posting costs and high exposure benefits, such as TripAdvisor and Yelp, will have more deception than communities with higher posting costs and comparatively lower exposure benefits, such as Orbitz.
- **Hypothesis 2**: Increasing the posting cost will reduce the prevalence of deception in a review community.

5.5 Experimental Setup

We apply the Bayesian Prevalence Model to reviews in six hotel review communities: Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor, and Yelp. We train our classifiers using the procedure given in Section 5.5.1. We then estimate each classifier's sensitivity and specificity using the procedure outlined in Section 5.5.2 and given in Appendix A.2. Gibbs sampling for the Bayesian Preva-

lence Model is performed using Equations A.1 and A.2 (given in Appendix A.1) for 70,000 iterations, with a *burn-in* of 20,000 iterations, and a *sampling lag* of 50. We use an uninformative (uniform) prior for π^*, i.e., $\boldsymbol{\alpha} = \langle 1, 1 \rangle$. Multiple runs are performed to verify the stability of the results.

5.5.1 Deception Classifier

The Bayesian Prevalence Model (Section 5.3) relies on the output of a deception classifier, which predicts whether an unlabeled review is DECEPTIVE or TRUTHFUL. Accordingly, we train supervised Support Vector Machine (SVM) classifiers on *n*-gram features (BIGRAMS$^+$), which performed well on our TripAdvisor hotel review experiments in Chapter 3.

As before, we train our classifier on 20 positive-sentiment DECEPTIVE hotel reviews from each of 20 Chicago hotels, as described in Section 2.3.1. However, in order to apply our prevalence model to review communities other than TripAdvisor, it is important that we train the underlying classifier on a broad sample of TRUTHFUL reviews from several review communities. Therefore, we pair the DECEPTIVE reviews with 20 positive-sentiment (5-star) TRUTHFUL reviews from each of the same 20 Chicago hotels, randomly sampled from six online hotel review communities (see Section 2.5.1), instead of just TripAdvisor.

5.5.2 Classifier Sensitivity and Specificity

The Bayesian Prevalence Model additionally exploits prior knowledge of the underlying deception classifier's deceptive recall rate, or *sensitivity* (η^*), and truthful recall rate, or *specificity* (θ^*). While it is not possible to obtain gold standard values for these parameters, we can obtain rough estimates of their values

Table 5.2: Reference DECEPTIVE recall (*sensitivity*) and TRUTHFUL recall (*specificity*) of an SVM classifier trained on reviews from six communities. Sensitivity is based on 5-fold cross-validation performance, while specificity is based on a random sample of reviews from each community.

Community	DECEPTIVE Recall (*sensitivity*)	TRUTHFUL Recall (*specificity*)
Expedia	86.8%	90.3%
Hotels.com	86.8%	88.5%
Orbitz	86.8%	89.8%
Priceline	86.8%	88.5%
TripAdvisor	86.8%	84.8%
Yelp	86.8%	68.5%

(denoted η and θ, respectively) through a combination of cross-validation and evaluation on a held-out development set.

In particular, because our training data contains gold standard DECEPTIVE reviews, we estimate the deceptive recall of our classifier by cross-validation on the training data. Unfortunately, our training data contains varying numbers of TRUTHFUL reviews from six review communities; accordingly, estimates of the truthful recall obtained by cross-validation may be biased towards communities with better representation in this training data. Therefore, we instead estimate the truthful recall of each community's classifier on a development set containing a random sample of 400 positive-sentiment (5-star) reviews from that community. We note that if the underlying (unknown) rate of deception is high among the reviews in this development set, then we may underestimate the specificity, and also the rate of deception for that community. We discuss this further in Section 5.6.1.

Reference sensitivity and specificity for each community is given in Table 5.2. We adopt an empirical Bayesian approach and use these estimates to inform the corresponding Beta priors via their hyperparameters, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, respectively. The

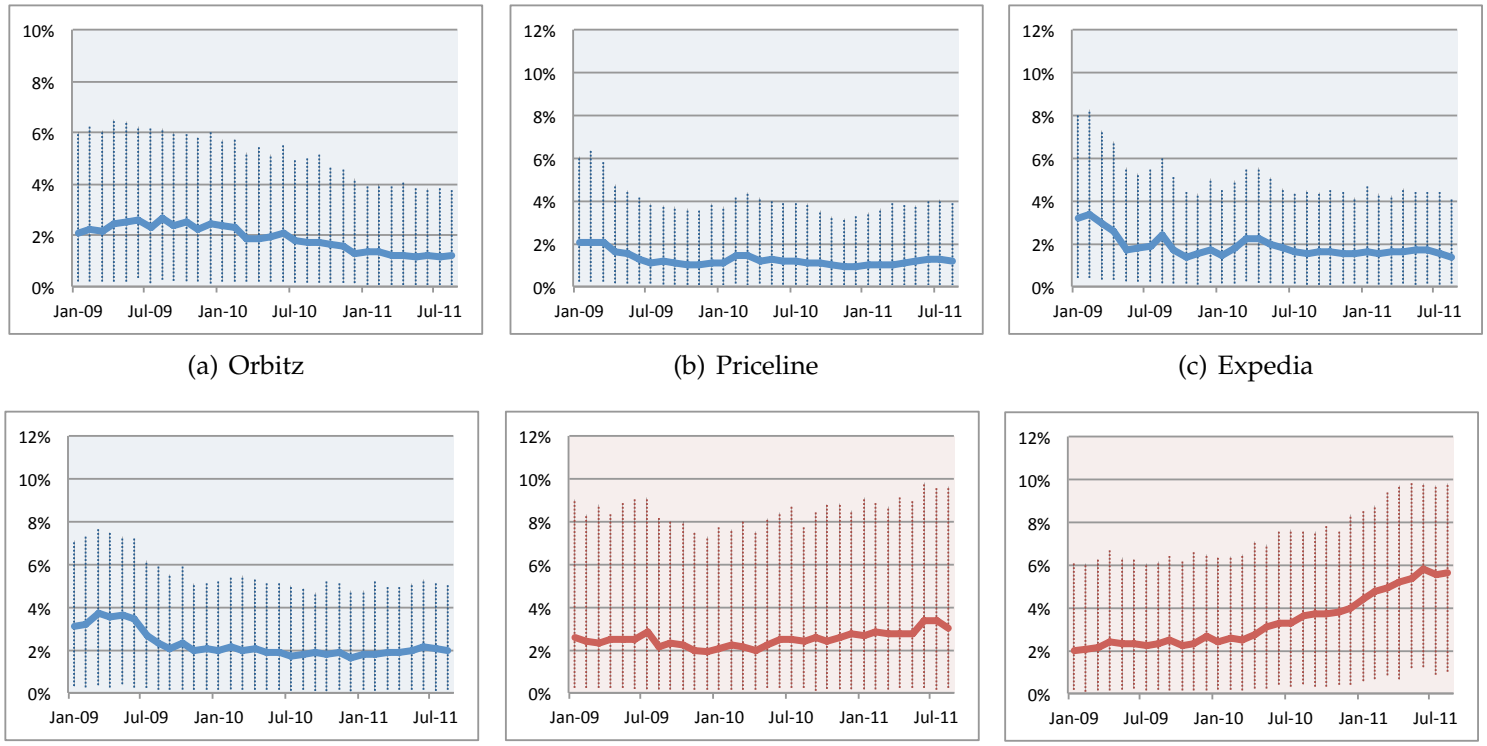
full procedure is given in Appendix A.2.

5.6 Results and Discussion

Estimates of the prevalence of deception over time in six review communities, obtained from the Bayesian Prevalence Model, are given in Figure 5.2. Blue graphs (a–d) correspond to communities with *High* posting cost (see Table 5.1), i.e., communities for which you are required to book a hotel room before posting a review, while red graphs (e–f) correspond to communities with *Low* posting cost, i.e., communities that allow any user to post reviews for any hotel.

Hypothesis 1 In agreement with *Hypothesis 1* (see Section 5.4.1), we observe that estimated rates of deception are either stationary or decreasing over time for *High* posting cost review communities (blue graphs, a–d). In contrast, we observe *growth* in the estimated rates of deception in review communities that allow any user to post reviews for any hotel, i.e., *Low* posting cost communities (red graphs, e–f). Interestingly, communities with a blend of posting costs appear to have intermediate rates of deception that are neither growing nor declining, e.g., Hotels.com, which has a steady estimated rate of deception $\approx 2\%$.

Hypothesis 2 Next, we test *Hypothesis 2*, i.e., that increasing a community's posting cost will decrease the prevalence of deception in that community. Unfortunately, we do not have the ability to increase any of the chosen community's real-world posting costs. Therefore, we instead simulate an increased posting cost by filtering reviews written by new users (i.e., first-time review writers), or inexperienced reviewers (i.e., first- or second-time review writers). In particular, by requiring users to post multiple reviews in order for any of



(a) Orbitz (b) Priceline (c) Expedia

(d) Hotels.com (e) Yelp (f) TripAdvisor

Figure 5.2: Bayesian estimates of deception prevalence versus time, for six online review communities. Blue (a–d) and red (e–f) graphs correspond to high and low posting cost communities, respectively. Error bars show Bayesian 95% credible intervals.

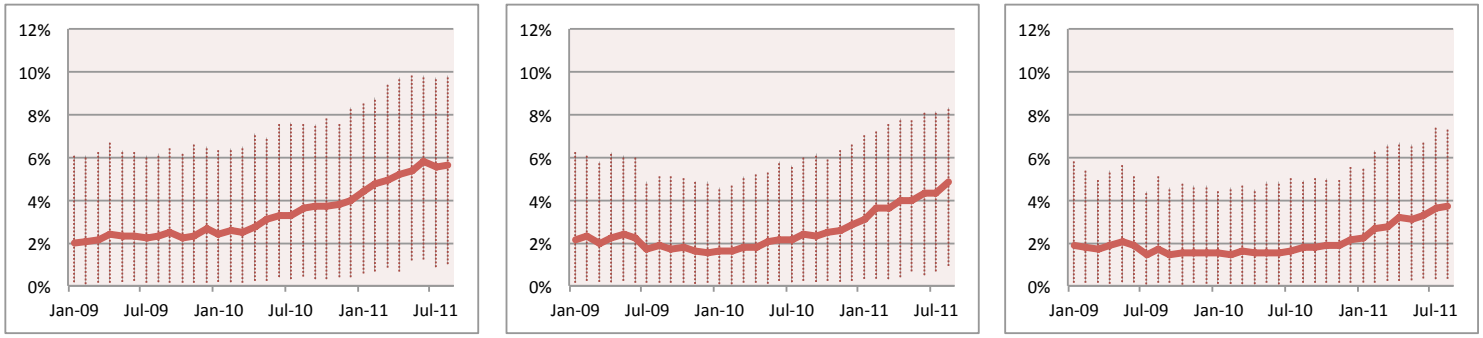
their reviews to be displayed, we are able to simulate a *retroactive* increase of a community's posting cost. Moreover, because this increased posting cost is simulated, this approach allows us to test our hypothesis without worrying about an "arms race" between spammers and review community operators.

We simulate this increased posting cost on our TripAdvisor data, because TripAdvisor is the only review community for which we know the number of reviews posted by each user. In particular, we apply our model to estimate the prevalence of deception over time after removing reviews written by first-time (*singleton*) review writers, and after removing reviews written by first- or second-time review writers. Note that reviews are filtered based on the author's number of TripAdvisor reviews as of February 2012, which is approximately six months after the end date of our dataset. Thus, at a minimum, authors of filtered reviews did not post a sufficient number of reviews to avoid the filtering criteria in the six months following their last review.

Bayesian Prevalence Model estimates for TripAdvisor for varying posting costs appear in Figure 5.3. In agreement with *Hypothesis 2*, we see a clear reduction in the prevalence of deception over time on TripAdvisor after removing these reviews, with rates dropping from $\approx 6\%$, to $\approx 5\%$, and finally to $\approx 4\%$, for each of the three settings, respectively. This suggests that an increased posting cost may be effective at reducing the prevalence of deception in online review communities.

5.6.1 Assumptions and Limitations

We have made a number of assumptions in this chapter, a few of which we will now highlight and discuss. Note that our choice of assumptions generally aim to underestimate, rather than overestimate, rates of deception.



(a) All reviews (b) Reviewers with ≤ 1 review excluded (c) Reviewers with ≤ 2 reviews excluded

Figure 5.3: Bayesian estimates of the prevalence of deception on TripAdvisor over time, when: (a) all reviews are included in the estimate; (b) reviews written by first-time (singleton) authors are excluded; and (c) reviews written by first- or second-time authors are excluded. Excluding reviews written by new users increases the posting cost, which is hypothesized to decrease the prevalence of deception.

First, we note that our unlabeled test set, containing reviews from the six online hotel review communities, overlaps with our TRUTHFUL training set. Consequently, we will *underestimate* the prevalence of deception, because the overlapping reviews will be more likely to be classified at test time as truthful, having been seen in training as being truthful. Excluding these overlapping reviews from the test set results in *overestimating* the prevalence of deception, based on the hypothesis that the overlapping reviews, chosen from 20 of the most popular Chicago hotels, are more likely to be truthful in the first place (see the discussion in Section 2.3.1).

Second, we observe that our development set, on which we estimate our classifier's truthful recall, is not gold standard. And while it is necessary to obtain a random sample of reviews in order to fairly estimate the classifier's truthful recall rate, such review samples are inherently unlabeled. This can be problematic if many of the reviews in our development set are, in fact, deceptive, because we will then *underestimate* our classifier's specificity, and therefore underestimate the prevalence of deception as well.

Third, our proposal for increasing the signal cost, by hiding reviews written by first- or second-time reviewers, is not perfect. While our results suggest that hiding these reviews will cause an immediate reduction in the prevalence of deception, the increase in posting cost may be insufficient to discourage new deception, once spammers become aware of the increased posting requirements.

Fourth, we have considered a limited version of the deception prevalence problem. In particular, we have restricted our analysis to positive Chicago hotel reviews, and our classifier is trained to detect only crowdsourced DECEPTIVE reviews. While these restrictions are necessary in this work, due to limitations of our dataset, future work may expand the methodology and analysis presented

in this chapter to other kinds of reviews, such as negative-sentiment reviews and deceptive reviews obtained from other sources (see other possible directions for future work in Section 6.2).

5.7 Chapter Summary

In this chapter, we have presented a general framework for estimating the prevalence of deception in online review communities, based on the output of a noisy deception classifier. Using this framework, we have explored the prevalence of deception among positive reviews in six popular online review communities, and provided the first empirical study of the magnitude, and influencing factors of deceptive opinion spam.

We have additionally proposed a theoretical model of online reviews as a signal to a product's true (unknown) quality. Specifically, we have argued that the prevalence of deception in a review community is a function of the posting costs and exposure benefits of that community. Based on this theory, we have further suggested two hypotheses, both of which are supported by our findings. First, we find that review communities with low posting costs and high exposure benefits have *more deception* than communities with comparatively higher posting costs or lower exposure benefits. Second, we find that by increasing the posting cost of a review community, i.e., by excluding reviews written by first- or second-time reviewers, we can effectively reduce both the prevalence and the growth rate of deception in that community.

CHAPTER 6

CONCLUSIONS AND FUTURE DIRECTIONS

In this dissertation, we have presented the first thorough investigation of deceptive opinion spam in online review communities. In this chapter, we summarize the contributions of this work (Section 6.1), and discuss possible directions for future work (Section 6.2).

6.1 Summary of Contributions

We have shown in Chapter 2 how deceptive opinions can be obtained quickly and cheaply through crowdsourcing services, such as Amazon Mechanical Turk, or from domain experts, such as hotel employees. We have additionally introduced, and made publicly available,[1] the first large-scale corpus of gold standard deceptive opinion spam, containing 1,280 deceptive reviews of hotels and restaurants, including a mix of positive-sentiment (4- or 5-star) and negative-sentiment (1- or 2-star) deceptive opinion spam, as well as domain expert deceptive opinion spam, written by hotel employees.

We have demonstrated in Chapter 3 that crowdsourced deceptive opinion spam is largely undetectable by human judges. We have also demonstrated that supervised Machine Learning classifiers can be trained to detect deceptive opinion spam, based on the language used in a review, with accuracies approaching 90% in some experiments. We have presented results, based on the feature weights learned by our classifiers, that illustrate the difficulties faced by liars in encoding spatial information into their deceptive reviews, and we have shown

[1]The data used in this dissertation is available from: http://www.cs.cornell.edu/~myleott/op_spam.

a plausible relationship between deceptive opinion spam and imaginative writing, based on part-of-speech distributional similarities.

In Chapter 4, we have shown that individual cues to deception vary according to the context of the review, including the sentiment and domain of the review, and the domain expertise of the reviewer. We have found that Machine Learning classifiers are sensitive to the context of the reviews on which they are trained, but that classifiers trained on reviews from several contexts perform better across contexts, compared to classifiers trained on reviews from a single context.

In Chapter 5, we introduced an approach for estimating the prevalence of deception in an online review community, based on the output of our Machine Learning classifiers. We have applied our approach to six online hotel review communities, and have presented the first empirical estimates of the prevalence of deceptive opinion spam among online hotel reviews. We have additionally shown that by increasing review posting costs, for example, by hiding reviews written by new users, we can reduce the prevalence of deceptive opinion spam.

6.2 Future Directions

This section discusses several possible directions for future work.

Context-Aware Deception Classifiers We have shown that a review's context influences cues to deception, and therefore, the ability of Machine Learning classifiers to detect deceptive opinion spam. Future work may develop additional techniques for handling context-specific features, potentially by directly modeling, or inferring, the context of the deception.

Reducing Deception Prevalence Future work might explore other methods for manipulating review posting costs, and the corresponding effects on deception prevalence. For example, some sites, such as Angie's List,[2] now charge a monthly access fee in order to browse or post reviews, and future work might study the effectiveness of such techniques at deterring deception.

Non-Experiential Product Reviews We have considered primarily reviews of experiential products, such as hotels and restaurants. Future work might additionally explore deception in non-experiential product review domains, such as product reviews found on Amazon.com.

Semi-Deceptive Opinion Spam There are many factors that may act to subtly influence or manipulate a user's opinion, besides the deceptions considered here. For example, many businesses attempt to coerce customers to post reviews with promises of discounts or other monetary incentives. Future work may try to model the influence that such incentives have on the resulting reviews.

Review Persuasiveness We have argued that online reviews are an important part of the modern consumer's decision making process. However, it seems unlikely that individual reviews are perceived as equally persuasive among consumers. For example, some websites now ask users to rate whether they find each review to be "helpful," but a more systematic evaluation of what makes a review persuasive, regardless of whether the review is truthful or deceptive, may be of benefit to businesses looking to better understand and improve their online reputations. Such an evaluation would additionally help to quantity the value of a review, and the benefit or harm caused by deceptive reviews.

[2] Angie's List: http://AngiesList.com.

APPENDIX A

ESTIMATING THE PREVALENCE OF DECEPTIVE OPINION SPAM

A.1 Gibbs Sampler for the Bayesian Prevalence Model

Gibbs sampling of the Bayesian Prevalence Model, introduced in Section 5.3, is performed according to the following conditional distributions:

$$\Pr(y_i = 1 \mid f(\mathbf{x}), \mathbf{y}^{(-i)}; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \propto (\boldsymbol{\alpha}_1 + N_1^{(-i)}) \cdot \frac{\boldsymbol{\beta}_{f(\mathbf{x}_i)} + X_{f(\mathbf{x}_i)}^{(-i)}}{\sum \boldsymbol{\beta} + N_1^{(-i)}}, \tag{A.1}$$

$$\Pr(y_i = 0 \mid f(\mathbf{x}), \mathbf{y}^{(-i)}; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \propto (\boldsymbol{\alpha}_0 + N_0^{(-i)}) \cdot \frac{\boldsymbol{\gamma}_{1-f(\mathbf{x}_i)} + Y_{f(\mathbf{x}_i)}^{(-i)}}{\sum \boldsymbol{\gamma} + N_0^{(-i)}}, \tag{A.2}$$

where,

$$\begin{aligned}
X_k^{(-i)} &= \sum_{j \neq i} \sigma[y_j = 1] \cdot \sigma[f(\mathbf{x}_j) = k], \\
Y_k^{(-i)} &= \sum_{j \neq i} \sigma[y_j = 0] \cdot \sigma[f(\mathbf{x}_j) = k], \\
N_1^{(-i)} &= X_0^{(-i)} + X_1^{(-i)}, \\
N_0^{(-i)} &= Y_0^{(-i)} + Y_1^{(-i)}.
\end{aligned}$$

After sampling, we reconstruct the collapsed variables to yield the Bayesian Prevalence Model estimate of the prevalence of deception:

$$\pi^* = \frac{\boldsymbol{\alpha}_1 + N_1}{\sum \boldsymbol{\alpha} + N^{\text{test}}}. \tag{A.3}$$

Estimates of the classifier's sensitivity and specificity are similarly given by:

$$\eta^* = \frac{\boldsymbol{\beta}_1 + X_1}{\sum \boldsymbol{\beta} + N_1}, \tag{A.4}$$

$$\theta^* = \frac{\boldsymbol{\gamma}_1 + Y_0}{\sum \boldsymbol{\gamma} + N_0}. \tag{A.5}$$

A.2 Estimating Classifier Sensitivity and Specificity

We estimate the sensitivity and specificity of our deception classifier via the following procedure:

1. Assume given a labeled training set, $\mathcal{D}^{\text{train}}$, containing N^{train} reviews of n hotels. Also assume given a development set, $\mathcal{D}^{\text{dev}}$, containing labeled truthful reviews.
2. Split $\mathcal{D}^{\text{train}}$ into n folds, $\mathcal{D}_1^{\text{train}}, \ldots, \mathcal{D}_n^{\text{train}}$, of sizes given by, $N_1^{\text{train}}, \ldots, N_n^{\text{train}}$, respectively, such that $\mathcal{D}_j^{\text{train}}$ contains all (and only) reviews of hotel j. Let $\mathcal{D}_{(-j)}^{\text{train}}$ contain all reviews *except* those of hotel j.
3. Then, for each hotel j:
   (a) Train a classifier, f_j, from reviews in $\mathcal{D}_{(-j)}^{\text{train}}$, and use it to classify reviews in $\mathcal{D}_j^{\text{train}}$.
   (b) Let $|TP|_j$ correspond to the observed number of true positives, i.e.:
   $$|TP|_j = \sum_{(\mathbf{x},y)\in\mathcal{D}_j^{\text{train}}} \sigma[y = 1] \cdot \sigma[f_j(\mathbf{x}) = 1]. \tag{A.6}$$
   (c) Similarly, let $|FN|_j$ correspond to the observed number of false negatives.
4. Calculate the aggregate number of true positives ($|TP|$) and false negatives ($|FN|$), and compute the sensitivity (deceptive recall) as:
   $$\eta = \frac{|TP|}{|TP| + |FN|}. \tag{A.7}$$
5. Train a classifier using *all* reviews in $\mathcal{D}^{\text{train}}$, and use it to classify reviews in $\mathcal{D}^{\text{dev}}$.

6. Let the resulting number of true negative and false positive predictions in $\mathcal{D}^{\text{dev}}$ be given by $|TN|_{dev}$ and $|FP|_{dev}$, respectively, and compute the specificity (truthful recall) as:

$$\theta = \frac{|TN|_{dev}}{|TN|_{dev} + |FP|_{dev}}. \tag{A.8}$$

For the Bayesian Prevalence Model, we observe that the posterior distribution of a variable with an uninformative (uniform) Beta prior, after observing a successes and b failures, is just Beta$(a + 1, b + 1)$, i.e., a and b are *pseudo counts*. Based on this observation, we set the hyperparameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, corresponding to the classifier's sensitivity (deceptive recall) and specificity (truthful recall), respectively, to:

$$\boldsymbol{\beta} = \langle |FN| + 1, |TP| + 1 \rangle,$$
$$\boldsymbol{\gamma} = \langle |FP|_{dev} + 1, |TN|_{dev} + 1 \rangle.$$

BIBLIOGRAPHY

[1] Cem Akkaya, Alexander Conrad, Janyce Wiebe, and Rada Mihalcea. Amazon Mechanical Turk for Subjectivity Word Sense Disambiguation. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 195–203. ACL, 2010.

[2] Michael Anderson and Jeremy Magruder. Learning from the crowd: Regression discontinuity estimates of the effects of an online review database. *The Economic Journal*, 2012.

[3] Sitaram Asur and Bernardo A. Huberman. Predicting the Future with Social Media. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 492–499. IEEE, 2010.

[4] Adam J. Berinsky, Gregory A. Huber, and Gabriel S. Lenz. Evaluating Online Labor Markets for Experimental Research: Amazon.com's Mechanical Turk. *Political Analysis*, 20(3):351–368, March 2012.

[5] Douglas Biber, Stig Johansson, Geoffrey Leech, Susan Conrad, Edward Finegan, and Randolph Quirk. *Longman Grammar of Spoken and Written English*. Pearson Education, 1999.

[6] Jeremy Birnholtz, Jamie Guillory, Jeff Hancock, and Natalya Bazarova. "on my way": Deceptive Texting and Interpersonal Awareness Narratives. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, pages 1–4. ACM, 2010.

[7] J. Pete Blair, Timothy R. Levine, and Allison S. Shaw. Content in Context Improves Deception Detection Accuracy. *Human Communication Research*, 36(3):423–442, June 2010.

[8] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, March 2011.

[9] Charles F. Bond and Bella M. DePaulo. Accuracy of deception judgments. *Personality and Social Psychology Review*, 10(3):214, 2006.

[10] Andrei Z. Broder. On the resemblance and containment of documents. In *Proceedings of Compression and Complexity of Sequences*, pages 21–29. IEEE, 1997.

[11] David B. Buller and Judee K. Burgoon. Interpersonal Deception Theory. *Communication Theory*, 6(3):203–242, 1996.

[12] John D. Burger, John Henderson, George Kim, and Guido Zarrella. Discriminating Gender on Twitter. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1301–1309. ACL, 2011.

[13] Whitney L. Cade, Blair A. Lehman, and Andrew Olney. An Exploration of Off Topic Conversation. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 669–672. ACL, 2010.

[14] Carlos Castillo, Debora Donato, Luca Becchetti, Paolo Boldi, Stefano Leonardi, Massimo Santini, and Sebastiano Vigna. A Reference Collection for Web Spam. *ACM SIGIR Forum*, 40(2):11–24, December 2006.

[15] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information Credibility on Twitter. In *Proceedings of the 20th International Conference on World Wide Web*, pages 675–684. ACM, 2011.

[16] Stanley F. Chen and Joshua Goodman. An Empirical Study of Smoothing Techniques for Language Modeling. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics, ACL, 1996.

[17] Munmun De Choudhury, S Counts, and Michael Gamon. Not All Moods Are Created Equal! Exploring Human Emotional States in Social Media. In *Proceedings of the 6th International AAAI Conference on Weblogs and Social Media*, pages 66–73. AAAI, 2012.

[18] Cone Communications. 2011 Online Influence Trend Tracker. URL: http://www.coneinc.com/negative-reviews-online-reverse-purchase-decisions, August 2011.

[19] Michael D. Conover, Bruno Gonçalves, Jacob Ratkiewicz, Alessandro Flammini, and Filippo Menczer. Predicting the political alignment of twitter users. In *Proceedings of the IEEE International Conference on Privacy, Security, Risk and Trust and Social Computing*. IEEE, 2011.

[20] "Crowdsourcing". Merriam-Webster.com. Retrieved June 16, 2013, from http://www.merriam-webster.com/dictionary/crowdsourcing.

[21] Aron Culotta. Towards detecting influenza epidemics by analyzing Twitter messages. In *Proceedings of the 1st Workshop on Social Media Analytics*, pages 115–122. ACM, 2010.

[22] Cristian Danescu-Niculescu-Mizil, Gueorgi Kossinets, Jon Kleinberg, and Lillian Lee. How Opinions are Received by Online Communities: A Case Study on Amazon.com Helpfulness Votes. In *Proceedings of the 18th International Conference on World Wide Web*, pages 141–150. ACM, 2009.

[23] Bella M. DePaulo, James J. Lindsay, Brian E. Malone, Laura Muhlenbruck, Kelly Charlton, and Harris Cooper. Cues to Deception. *Psychological Bulletin*, 129(1):74–118, 2003.

[24] Harris Drucker, Donghui Wu, and Vladimir N. Vapnik. Support Vector Machines for Spam Categorization. *IEEE Transactions on Neural Networks*, 10(5):1048–1054, 1999.

[25] Jacob Eisenstein, Amr Ahmed, and Eric P. Xing. Sparse Additive Generative Models of Text. In *Proceedings of the 28th International Conference on Machine Learning*, pages 1041–1048, 2011.

[26] Jacob Eisenstein, Brendan OConnor, Noah A. Smith, and Eric P. Xing. A Latent Variable Model for Geographic Lexical Variation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1277–1287. ACL, 2010.

[27] Jacob Eisenstein, Noah A. Smith, and Eric P. Xing. Discovering Sociolinguistic Associations with Structured Sparsity. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 1365–1374. ACL, 2011.

[28] Paul Ekman and Wallace V. Friesen. Nonverbal Leakage and Clues to Deception. *Psychiatry*, 32(1):88, 1969.

[29] Federal Trade Commission. Guides Concerning Use of Endorsements and Testimonials in Advertising. *FTC 16 CFR Part 255*, 2009.

[30] Thomas H. Feeley and Mark A. DeTurck. The behavioral correlates of sanctioned and unsanctioned deceptive communication. *Journal of Nonverbal Behavior*, 22(3):189–204, 1998.

[31] Song Feng, Ritwik Banerjee, and Yejin Choi. Characterizing Stylistic Ele-

ments in Syntactic Structure. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1522–1533. ACL, 2012.

[32] Song Feng, Ritwik Banerjee, and Yejin Choi. Syntactic Stylometry for Deception Detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers*, pages 171–175. ACL, 2012.

[33] Song Feng, Longfei Xing, Anupam Gogar, and Yejin Choi. Distributional Footprints of Deceptive Product Reviews. In *Proceedings of the 6th International AAAI Conference on Weblogs and Social Media*, pages 98–105. AAAI, 2012.

[34] George Forman and Martin Scholz. Apples-to-Apples in Cross-Validation Studies: Pitfalls in Classifier Performance Measurement. *ACM SIGKDD Explorations Newsletter*, 12(1):49–57, 2009.

[35] Mark G. Frank and Paul Ekman. The Ability To Detect Deceit Generalizes Across Different Types of High-Stake Lies. *Journal of Personality and Social Psychology*, 72(6):1429–1439, 1997.

[36] Daniel Gayo-Avello. "I Wanted to Predict Elections with Twitter and all I got was this Lousy Paper": A Balanced Survey on Election Prediction using Twitter Data. *arXiv:1204.6441 [cs.CY]*, pages 1–13, 2012.

[37] Eric Gilbert and Karrie Karahalios. Widespread Worry and the Stock Market. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, pages 59–65. AAAI, 2010.

[38] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating Web Spam with TrustRank. In *Proceedings of the 30th International Conference on Very Large Data Bases*, pages 576–587. VLDB Endowment, 2004.

[39] Jeffrey T. Hancock, Lauren E. Curry, Saurabh Goorha, and Michael Woodworth. On Lying and Being Lied To: A Linguistic Analysis of Deception in Computer-Mediated Communication. *Discourse Processes*, 45(1):1–23, 2007.

[40] Jeffrey T. Hancock, Jennifer Thom-Santelli, and Thompson Ritchie. Deception and Design: The Impact of Communication Technology on Lying Behavior. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 129–134. ACM, 2004.

[41] Rahaf Harfoush. *Yes We Did!: An Inside Look at How Social Media Built the Obama Brand*. New Riders Pub, 2009.

[42] Thorsten Hennig-Thurau, Kevin P. Gwinner, Gianfranco Walsh, and Dwayne D. Gremler. Electronic word-of-mouth via consumer-opinion platforms: What motivates consumers to articulate themselves on the Internet? *Journal of Interactive Marketing*, 18(1):38–52, January 2004.

[43] Liangjie Hong, Amr Ahmed, Siva Gurumurthy, Alex Smola, and Kostas Tsioutsiouliklis. Discovering Geographical Topics In The Twitter Stream. In *Proceedings of the 21st international conference on World Wide Web*, page 769. ACM, 2012.

[44] Nan Hu, Ling Liu, and Jie Jennifer Zhang. Do online reviews affect product sales? The role of reviewer characteristics and temporal effects. *Information Technology and Management*, 9(3):201–214, 2008.

[45] Panos Ipeirotis. Demographics of Mechanical Turk. *NYU Working Paper No. CEDER-10-01*, 2010.

[46] Ipsos. Socialogue: Five Stars? Thumbs Up? A+ or Just Average? URL: http://www.ipsos-na.com/news-polls/pressrelease.aspx?id=5929, 2012.

[47] Tom N. Jagatic, Nathaniel A. Johnson, Markus Jakobsson, and Filippo Menczer. Social phishing. *Communications of the ACM*, 50(10):94–100, 2007.

[48] Nitin Jindal and Bing Liu. Opinion spam and analysis. In *Proceedings of the International Conference on Web Search and Web Data Mining*, pages 219–230. ACM, 2008.

[49] Thorsten Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In *Proceedings of the 10th European Conference on Machine Learning*, pages 137–142, 1998.

[50] Thorsten Joachims. Making large-scale support vector machine learning practical. In *Advances in Kernel Methods - Support Vector Learning*, pages 169–184. MIT Press, 1999.

[51] Marcia K. Johnson and Carol L. Raye. Reality monitoring. *Psychological Review*, 88(1):67–85, 1981.

[52] Marcia K. Johnson and Carol L. Raye. False memories and confabulation. *Trends in Cognitive Sciences*, 2(4):137–45, April 1998.

[53] Wesley O. Johnson, Joseph L. Gastwirth, and Larry M. Pearson. Screening without a "Gold Standard": The Hui-Walter Paradigm Revisited. *American Journal of Epidemiology*, 153(9):921, 2001.

[54] Lawrence Joseph, Theresa W. Gyorkos, and Louis Coupal. Bayesian Estimation of Disease Prevalence and the Parameters of Diagnostic Tests in the Absence of a Gold Standard. *American Journal of Epidemiology*, 141(3):263, 1995.

[55] Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 423–430. Association for Computational Linguistics, ACL, 2006.

[56] Dan Klein and Christopher D. Manning. Accurate Unlexicalized Parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 423–430. ACL, 2003.

[57] Moshe Koppel, Shlomo Argamon, and Anat Rachel Shimoni. Automatically Categorizing Written Texts by Author Gender. *Literary and Linguistic Computing*, 17(4):401–412, 2002.

[58] Moshe Koppel, Jonathan Schler, and Kfir Zigdon. Determining an author's native language by mining a text for errors. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pages 624–628. ACM, 2005.

[59] J. Richard Landis and Gary G. Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174, 1977.

[60] Mira Lee, Shelly Rodgers, and Mikyoung Kim. Effects of Valence and Extremity of eWOM on Attitude toward the Brand and Website. *Journal of Current Issues & Research in Advertising*, 31(2):1–11, September 2009.

[61] Timothy R. Levine, Rachel K. Kim, and J. Pete Blair. (In)accuracy at Detecting True and False Confessions and Denials: An Initial Test of a Projected Motive Model of Veracity Judgments. *Human Communication Research*, 36(1):82–102, 2010.

[62] Fangtao Li, Minlie Huang, Yi Yang, and Xiaoyan Zhu. Learning to identify review spam. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 2488–2493. AAAI, 2011.

[63] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady W. Lauw. Detecting Product Review Spammers using Rating Behaviors. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 939–948. ACM, 2010.

[64] Bin Lu, Myle Ott, Claire Cardie, and Benjamin K. Tsou. Multi-aspect Sentiment Analysis with Topic Models. In *Proceedings of the 11th International Conference on Data Mining: Workshops*, pages 81–88, Vancouver, British Columbia, Canada, December 2011. IEEE.

[65] Michael Luca. Reviews, Reputation, and Revenue: The Case of Yelp.com. *Harvard Business School Working Paper, No. 12-016*, 2011.

[66] François Mairesse, Marilyn A. Walker, Matthias R. Mehl, and Roger K. Moore. Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text. *Journal of Artificial Intelligence Research*, 30(1):457–500, 2007.

[67] Dina Mayzlin, Yaniv Dover, and Judith Chevalier. Promotional Reviews: An Empirical Investigation of Online Review Manipulation. *NBER Working Paper No. 18340*, 2012.

[68] Marcelo Mendoza, Barbara Poblete, and Carlos Castillo. Twitter Under Crisis: Can we trust what we RT? In *Proceedings of the 1st Workshop on Social Media Analytics*, pages 71–79. ACM, 2010.

[69] Rohith K. Menon and Yejin Choi. Domain Independent Authorship Attribution without Domain Adaptation. In *Proceedings of Recent Advances in Natural Language Processing*, pages 309–315. ACL, 2011.

[70] Panagiotis T. Metaxas and Eni Mustafaraj. From Obscurity to Prominence in Minutes: Political Speech and Real-Time Search. In *Proceedings of WebSci10: Extending the Frontiers of Society*, pages 1–7, 2010.

[71] Panagiotis T. Metaxas and Eni Mustafaraj. Social Media and the Elections. *Science*, 338(6106):472–473, 2012.

[72] David Meyer. Fake reviews prompt Belkin apology. URL: http://news.cnet.com/8301-1001_3-10145399-92.html, 2009.

[73] Rada Mihalcea and Carlo Strapparava. The Lie Detector: Explorations in the Automatic Recognition of Deceptive Language. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 309–312. ACL and AFNLP, 2009.

[74] Gilad Mishne, David Carmel, and Ronny Lempel. Blocking Blog Spam with Language Model Disagreement. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web*, pages 1–6, 2005.

[75] Meredith R. Morris, Scott Counts, Asta Roseway, Aaron Hoff, and Julia Schwarz. Tweeting is Believing? Understanding Microblog Credibility Perceptions. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, pages 441–450. ACM, 2012.

[76] Arjun Mukherjee and Bing Liu. Improving Gender Classification of Blog Authors. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 207–217. ACL, 2010.

[77] Arjun Mukherjee, Bing Liu, and Natalie Glance. Spotting Fake Reviewer Groups in Consumer Reviews. In *Proceedings of the 21st international conference on World Wide Web*, pages 191–200. ACM, 2012.

[78] Matthew L. Newman, James W. Pennebaker, Diane S. Berry, and Jane M. Richards. Lying Words: Predicting Deception From Linguistic Styles. *Personality and Social Psychology Bulletin*, 29(5):665–675, 2003.

[79] Andrew Y. Ng and Michael I. Jordan. On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes. In *Proceedings of Advances in Neural Information Processing Systems*, pages 841–848, 2002.

[80] Dong Nguyen, Noah A. Smith, and Carolyn P. Rosé. Author Age Prediction from Text using Linear Regression. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 115–123. ACL, 2011.

[81] Alexandru Niculescu-Mizil and Rich Caruana. Predicting Good Probabilities With Supervised Learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 625–632. ACM, 2005.

[82] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting Spam Web Pages through Content Analysis. In *Proceedings of the 15th International Conference on World Wide Web*, pages 83–92. ACM, 2006.

[83] Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, pages 122–129. AAAI, 2010.

[84] Michael P. O'Mahony and Barry Smyth. Learning to Recommend Helpful Hotel Reviews. In *Proceedings of the 3rd ACM Conference on Recommender Systems*, pages 305–308. ACM, 2009.

[85] Beng Soo Ong. The Perceived Influence of User Reviews in the Hospitality Industry. *Journal of Hospitality Marketing & Management*, 21(5):463–485, July 2012.

[86] Myle Ott, Claire Cardie, and Jeff Hancock. Estimating the Prevalence of Deception in Online Review Communities. In *Proceedings of the 21st International Conference on World Wide Web*, pages 201–210. ACM, 2012.

[87] Myle Ott, Claire Cardie, and Jeffrey T. Hancock. Negative Deceptive Opinion Spam. In *Proceedings of NAACL-HLT 2013*, pages 497–501. ACL, 2013.

[88] Myle Ott, Yejin Choi, Cardie Cardie, and Jeffrey T. Hancock. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 309–319. ACL, 2011.

[89] Sinno Jialin Pan and Qiang Yang. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, October 2010.

[90] Bo Pang and Lillian Lee. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(12):1–135, 2008.

[91] Cheol Park and Thae Min Lee. Information direction, website reputation and eWOM effect: A moderating role of product type. *Journal of Business Research*, 62(1):61–67, January 2009.

[92] Michael J. Paul and Mark Dredze. You Are What You Tweet: Analyzing Twitter for Public Health. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*, pages 265–272. AAAI, 2011.

[93] F Peng, D Schuurmans, S Wang, and V Keselj. Language independent authorship attribution using character level language models. *Proceedings of the tenth . . .* , pages 267–274, 2003.

[94] Fuchun Peng and Dale Schuurmans. Combining Naive Bayes and n-Gram Language Models for Text Classification. In *Proceedings of the 25th European Conference on IR Research*, pages 335–350. Springer, 2003.

[95] Marco Pennacchiotti and Ana-Maria Popescu. A Machine Learning Approach to Twitter User Classification. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*, pages 281–288. AAAI, 2011.

[96] James W. Pennebaker, Cindy K. Chung, Molly Ireland, Amy Gonzales, and Roger J. Booth. The Development and Psychometric Properties of LIWC2007. *LIWC.net*, 2007.

[97] James W. Pennebaker, Matthias R. Mehl, and Kate G. Niederhoffer. Psychological Aspects of Natural Language Use: Our Words, Our Selves. *Annual Review of Psychology*, 54(1):547–77, January 2003.

[98] Stephen Porter and John C. Yuille. The Language of Deceit: An Investigation of the Verbal Clues to Deception in the Interrogation Context. *Law and Human Behavior*, 20(4):443–458, 1996.

[99] Martin Potthast, Tim Gollub, Matthias Hagen, Jan Graßegger, Johannes Kiesel, Maximilian Michel, Arnd Oberländer, Martin Tippmann, Alberto Barrón-Cedeno, Parth Gupta, Paolo Rosso, and Benno Stein. Overview of the 4th International Competition on Plagiarism Detection. In *CLEF 2012 Evaluation Labs and Workshop Working Notes Papers*, 2012.

[100] Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, and Qiaozhu Mei. Rumor has it: Identifying Misinformation in Microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599. ACL, 2011.

[101] Erik Qualman. *Socialnomics: How social media transforms the way we live and do business*. John Wiley & Sons, 2012.

[102] Sindhu Raghavan, Adriana Kovashka, and Raymond Mooney. Authorship Attribution Using Probabilistic Context-Free Grammars. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 38–42. ACL, 2010.

[103] Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. Classifying Latent User Attributes in Twitter. In *Proceedings of the 2nd International Workshop on Search and Mining User-generated Contents*, pages 37–44. ACM, 2010.

[104] Jacob Ratkiewicz, Michael D. Conover, Mark Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer. Detecting and Tracking Political Abuse in Social Media. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*, pages 297–304. AAAI, 2011.

[105] Paul Rayson, Andrew Wilson, and Geoffrey Leech. Grammatical word class variation within the British National Corpus sampler. *Language and Computers*, 36(1):295–306, 2001.

[106] Jason D. M. Rennie and Ryan Rifkin. Improving Multiclass Text Classification with the Support Vector Machine. *Massachusetts Institute of Technology AI Memo 2001-026*, pages 1–14, 2001.

[107] Ryan Rifkin and Aldebaro Klautau. In Defense of One-Vs-All Classification. *The Journal of Machine Learning Research*, 5:101–141, 2004.

[108] Robert A. Rigby and D. Mikis Stasinopoulos. Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54(3):507–554, 2005.

[109] Joel Ross, Lilly Irani, M. Six Silberman, Andrew Zaldivar, and Bill Tomlinson. Who are the Crowdworkers? Shifting Demographics in Mechanical Turk. In *Proceedings of CHI '10 Extended Abstracts on Human Factors in Computing Systems*, pages 2863–2872. ACM, 2010.

[110] Eduardo J. Ruiz, Vagelis Hristidis, Carlos Castillo, Aristides Gionis, and Alejandro Jaimes. Correlating Financial Time Series with Micro-Blogging Activity. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*, pages 513–522. ACM, 2012.

[111] Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the Outputs of a Classifier to New a Priori Probabilities: A Simple Procedure. *Neural computation*, 14(1):21–41, January 2002.

[112] Ruchita Sarawgi, Kailash Gajulapalli, and Yejin Choi. Gender Attribution: Tracing Stylometric Evidence Beyond Topic and Genre. In *Proceedings of the 15th Conference on Computational Natural Language Learning*, pages 78–86. ACL, 2011.

[113] David O. Sears. College Sophomores in the Laboratory: Influences of a Narrow Data Base on Social Psychology's View of Human Nature. *Journal of Personality and Social Psychology*, 51(3):515–530, 1986.

[114] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[115] M. Ángeles Serrano, Alessandro Flammini, and Filippo Menczer. Modeling Statistical Properties of Written Text. *PloS one*, 4(4):5372, 2009.

[116] John J. Skowronski and Donal E. Carlston. Negativity and Extremity Biases in Impression Formation: A Review of Explanations. *Psychological Bulletin*, 105(1):131–142, 1989.

[117] Michael Spence. Job Market Signaling. *The Quarterly Journal of Economics*, 87(3):355–374, 1973.

[118] Andreas Stolcke. SRILM - An Extensible Language Modeling Toolkit. In *Proceedings of the 7th International Conference on Spoken Language Processing*, pages 901–904, 2002.

[119] Bradley Taylor, Dan Fingal, and Douglas Aberdeen. The War Against Spam: A report from the front line. In *Proceedings of the NIPS Workshop on Machine Learning in Adversarial Environments*, pages 1–3, 2007.

[120] Yee Whye Teh. A Hierarchical Bayesian LanguageModel based on Pitman-Yor Processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992. ACL, 2006.

[121] Catalina L. Toma and Jeffrey T. Hancock. What Lies Beneath: The Linguistic Traces of Deception in Online Dating Profiles. *Journal of Communication*, 62(1):78–97, 2012.

[122] Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner, and Isabell M. Welpe. Predicting Elections with Twitter: What 140 Characters

Reveal about Political Sentiment. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, pages 178–185. AAAI, 2010.

[123] Sudhir Varma and Richard Simon. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 7(1):91, January 2006.

[124] Aldert Vrij. *Detecting Lies and Deceit: Pitfalls and Opportunities*. Wiley, 2008.

[125] Aldert Vrij, Sharon Leal, Pär Anders Granhag, Samantha Mann, Ronald P. Fisher, Jackie Hillman, and Kathryn Sperry. Outsmarting the Liars: The Benefit of Asking Unanticipated Question. *Law and Human Behavior*, 33(2):159–166, 2009.

[126] Aldert Vrij, Samantha Mann, Susanne Kristen, and Ronald P. Fisher. Cues to Deception and Ability to Detect Lies as a Function of Police Interview Styles. *Law and Human Behavior*, 31(5):499–518, 2007.

[127] William Warner and Julia Hirschberg. Detecting hate speech on the world wide web. In *Proceedings of the 2012 Workshop on Language in Social Media*, pages 19–26. ACL, 2012.

[128] Wouter Weerkamp and Maarten De Rijke. Credibility Improves Topical Blog Post Retrieval. *ACL-08: HLT*, pages 923–931, 2008.

[129] Markus Weimer, Iryna Gurevych, and Max Mühlhäuser. Automatically Assessing the Post Quality in Online Discussions on Software. In *Proceedings of the ACL 2007 Demo and Poster Sessions*, pages 125–128. Association for Computational Linguistics, ACL, 2007.

[130] Felix Ming Fai Wong, Soumya Sen, and Mung Chiang. Why Watching Movie TweetsWont Tell the Whole Story? In *Proceedings of the 2012 ACM Workshop on Online Social Networks*, pages 61–66. ACM, 2012.

[131] Guangyu Wu, Derek Greene, and Pádraig Cunningham. Merging Multiple Criteria to Identify Suspicious Reviews. In *Proceedings of the 4th ACM Conference on Recommender Systems*, pages 241–244. ACM, 2010.

[132] Guangyu Wu, Derek Greene, Barry Smyth, and Pádraig Cunningham. Distortion as a Validation Criterion in the Identification of Suspicious Reviews. In *Proceedings of the 1st Workshop on Social Media Analytics*, pages 10–13. ACM, 2010.

[133] Guang Xiang, Bin Fan, Ling Wang, Jason Hong, and Carolyn P. Rose. Detecting Offensive Tweets via Topical Feature Discovery over a Large Scale Twitter Corpus. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 1980–1984. ACM, 2012.

[134] Kyung-Hyan Yoo and Ulrike Gretzel. Comparison of Deceptive and Truthful Travel Reviews. *Information and Communication Technologies in Tourism*, pages 37–47, 2009.

[135] Sheng Yu and Subhash Kak. A Survey of Prediction Using Social Media. *arXiv preprint arXiv:1203.1647*, pages 1–20, 2012.

[136] Lina Zhou, Judee K. Burgoon, Douglas P. Twitchell, Tiantian Qin, and Jay F. Nunamaker. A comparison of classification methods for predicting deception in computer-mediated communication. *Journal of Management Information Systems*, 20(4):139–166, 2004.

[137] Lina Zhou, Yongmei Shi, and Dongsong Zhang. A Statistical Language Modeling Approach to Online Deception Detection. *IEEE Transactions on Knowledge and Data Engineering*, 20(8):1077–1081, 2008.

[138] Feng Zhu and Xiaoquan Zhang. Impact of Online Consumer Reviews on Sales: The Moderating Role of Product and Consumer Characteristics. *Journal of Marketing*, 74(2):133–148, 2010.