UNIVERSITY OF CALIFORNIA,
IRVINE

An Analysis of Deceptive Text Using Techniques of Machine Learning, Corpus Generation, and Online Crowdsourcing

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Cognitive Sciences

by

Dan Barsever

Dissertation Committee:
Assistant Professor Emre Neftci, Chair
Professor Mark Steyvers
Associate Professor Sameer Singh

2022

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank the Department of Cognitive Sciences for their support. I am unendingly appreciate of all of the instruction and administrative efforts on my behalf.

I would to thank my mother, father, and sister, who have helped and supported me on this journey.

I would like to thank all the members of NMI Lab and MADLAB, whose community has been a constant source of aid.

I would like to thank my committee, Dr. Mark Steyvers and Dr. Sameer Singh, for their encouragement, insight, and support.

I especially want to thank my advisor, Dr. Emre Neftci. Without his guidance and wisdom throughout these years of study, none of this research would be possible.

VITA

Dan Barsever

EDUCATION

Doctor of Philosophy in Cognitive Sciences **2022**
University of California, Irvine *Irvine, California*

Bachelor of Science in Electrical Engineering **2016**
University of California, Irvine *Irvine, California*

RESEARCH EXPERIENCE

Graduate Research Assistant **2016–2022**
University of California, Irvine *Irvine, California*

TEACHING EXPERIENCE

Teaching Assistant **2016–2021**
University of California, Irvine *Irvine, California*

REFEREED JOURNAL PUBLICATIONS

Building and Benchmarking the Motivated Deception Corpus: Improving the Quality of Deceptive Text Through Gaming — **Under Review**
Behavior Research Methods

REFEREED CONFERENCE PUBLICATIONS

Building a Better Lie Detector with BERT: The Difference Between Truth and Lies — **July 2020**
IEEE International Joint Conference on Neural Networks (IJCNN)

ABSTRACT OF THE DISSERTATION

An Analysis of Deceptive Text Using Techniques of Machine Learning, Corpus Generation, and Online Crowdsourcing

By

Dan Barsever

Doctor of Philosophy in Cognitive Sciences

University of California, Irvine, 2022

Assistant Professor Emre Neftci, Chair

This research demonstrates how to use deep learning techniques alongside corpus generation and online crowdsourcing in order to better understand deceptive text. In this dissertation, I use state-of-the-art classifiers to examine the structure of deceptive text and determine what parts mark it as deceptive. I also expand knowledge of deception into new areas by adding to the knowledge base of deceptive text with large amounts of curated, realistic data. It also offers a more complete understanding by examining deceptive text through multiple lenses. My research accomplishes this through three interrelated projects: (I) The construction of a new state-of-the-art classifier, and modifying the input to the classifier to examine what the classifier considers most informative in a classification, (II) the creation of a new corpus of deceptive text, the Motivated Deception Corpus, which uses gameifying techniques to improve the quality of deceptive text samples by making them more realistic through competition, and (III) a human subject study on Amazon Mechanical Turk, where I observe what samples humans consider deceptive or truthful and use a Cultural Consensus Theory model to identify what prompts a subject to decide one way or the other.

Chapter 1

Introduction

1.1 The History of Deception

Deception has been an integral part of social interaction for as long as communication has existed. Concurrently, detecting deception has been a crucial skill for equally as long. We perform it when a friend says they're too busy to go to a party when they're really just tired, when an employee claims to be sick can't come in to work, when a politician promises that *this* time they'll vote the way that you want them to. For every medium of communication, there is a commensurate method by which people lie. And as communication has evolved, so too has the methods by which deception is performed and detected.

As the internet has evolved and social media become a larger part of daily life, deception through the medium of text has become more prevalent and relevant. However, classic techniques of lie detection do not easily transfer over to text. Most traditional methods of lie detection consist of analyzing a physiological response, such as sweat or heart rate. One of the oldest techniques in fact originates in ancient China, where an interrogated suspect would be forced to hold dry rice in their mouth while being questioned [Ford, 2006]. If the

rice remained dry after the questioning, they were held to be lying, since dry mouth was considered a symptom of lying.

One of the most common incarnations of colloquial lie detection is the polygraph [Council et al., 2003]. At it's most basic it's a system that examines physiological responses such as increased sweat or heart rate that are expected to occur when people lie. Audio-visual analysis of voices and facial expressions is also popular, and has inspired datasets such as the video-based database of deception gathered by Lloyd et al which contains video recordings of people of different genders and ethnicities speaking either honestly or dishonestly about their social relationships [Lloyd et al., 2019]. On the more verbal side of the spectrum are the analyses described by Fitzpatrick et al, , who focus on clusters of features present in verbal—as well as physiological and gestural—behaviors [Fitzpatrick et al., 2015]. They too, note the difference between data collected in a laboratory and the "real world." There is also the work done by Aboueleniun et al, which incorporates physiological, linguistic, and thermal recordings to create a more accurate deception detection system [Abouelenien et al., 2016]. For the most part, these approaches are ineffective in text, where there are no physiological clues [Ott et al., 2011], but they establish how it is possible to use data of multiple types to improve detection accuracy. By combining different mediums, it is possible to increase the amount of features that can be watched for, and therefore increase discrimination accuracy.

Data on deceptive text is limited, which is to be expected given the uncertain nature of the text. A popular gold-standard dataset is the Ott Deceptive Opinion Spam dataset, which consists of 1600 true and false hotel reviews sourced from TripAdvisor and Amazon Mechanical Turk [Ott et al., 2011]. In the realm of natural language processing, this is a miniscule amount of data. Compare the Ott dataset to the Twitter-based sentiment corpus from Pak and Paroubek, which contains 300,000 samples [Pak and Paroubek, 2010].

The problem is that this dearth of data is not easy to fix. Supervised learning techniques tend to suffer from a problem known as sample inefficiency. Before data can be used in

a machine learning classifier, it must be labeled. This labeling is often done by a human judge, and tends to be an expensive, painstaking task. When possible, labelers exploit existing labels; for example if a movie review has a 'thumbs up' marker, it can generally be labeled as positive. This is not a solution that works for deceptive text since, unlike a sentiment task, self-reported labels of deception are rare. Making matters worse is that deceptive text cannot easily be verified by a human, which means samples must be gathered carefully and are not simple to scrape from large data. This leads to a relative drought in labeled data even among other supervised learning tasks.

1.2 The Importance of Text

Being one of the most prolific means of communication, whether it be in phone messages, forum posts, or magazine articles, text contains many ways to deceive the reader. Why do we care about text in particular? Because textual information informs real-life decisions. We make purchase decisions based on reviews we read about products [Local, 2018]. We make political decisions based on what we read on social media [Highfield, 2017]). And while we can easily tell whether a review is positive or negative, determining whether that review is truthful is another matter entirely.

One example of widespread, influential deceptive text that has risen to prominence is fake news. This usually takes the form of misinformation or disinformation presented as fact, either on a traditional news source or a social media source such as Facebook. While not a new phenomenon, the subject of fake news has come under increased scrutiny in recent years [Kalsnes, 2018]. Fake news, misinformation, or even simply the fear of it can influence people's perceptions of current events, which can influence their political and social views. It is especially prevalent on social media, where the nature of the medium, such as the short time between event and reporting as well as the diversity of the subject matter, makes

detecting falsity a singularly difficult challenge [Shu et al., 2017]. Some datasets, like the one developed by Tasnim et al, try to combat this problem by including contextual information and spatiotemporal data [Tasnim et al., 2020]. These false stories impact consumers directly by influencing decisions made that affect the reader's own life. For example, populations that are subject to misinformation about crucial vaccines such as the COVID-19 vaccine become less likely to inoculate themselves and their dependents, which can have far-reaching effects [Carrieri et al., 2019].

More than our political views, deceptive text can affect financial decisions that have an immediate and personal impact on a consumer. What does one do when deciding to buy a new car, or selecting which movie to see on an outing? Many people turn to public reviews, where other consumers, or sometimes experts in the field, leave their opinion on the quality of the item you might spend your money on. If the reviews are positive, that can cause someone to take the risk and part with their money. If they are negative, that can make a prospective buyer shun that product or seller and refuse to do business [Local, 2018].

Reviews make up a truly massive amount of text data, with the Amazon Customer Reviews Dataset alone comprising over a hundred million reviews [Amazon, 2014]. Problems arise, however, when these reviews are not truthful. This usually takes the form of a malicious customer posting fake negative reviews to hurt a business, or a company shill posting fake positive reviews to inflate its image. Fournaciari et al observed this type of review in the plethora of these so-called "sock-puppet" reviews of books that were in truth written by the book's author to drum up sales [Fornaciari and Poesio, 2014]. They also note the difficulty in labeling 'real' deceptive samples, and how they were forced to identify cues that they believed indicated deception without being able to know the absolute ground truth. This is a common problem because humans are ineffective at detecting deceptive text, faring little better than chance [Levine, 2014, Ott et al., 2013].

This is a perfectly logical and prudent approach, but it comes with additional risks when

you factor in false reviews, also known as Deceptive Opinion Spam. This can take several different forms, affecting both seller and buyers negatively. A malicious user can post a negative review of a company they don't like, regardless of whether or not it is a true opinion, in the hope of driving away other potential customers and hurting that business. On the flip side, the company itself can post an erroneous positive review of their own product. Doing this can entice people who would otherwise avoid them into buying a likely inferior product. When almost ninety percent of consumers read online reviews, this can end up affecting large amounts of money [Local, 2018].

Complicating the issue is that humans are ineffective at detecting deceptive text. While people may have at least a cursory understanding of lie detection in person, when it comes to text most people fare little better than chance [Levine, 2014, Ott et al., 2013]. This is in stark contrast to other linguistic tasks such as sentiment analysis (e.g. identifying if a text sample is praising or condemning something) where humans perform extremely well [Vogler and Pearl, 2019].

1.3 Human Performance

How well humans perform deception detection depends on many factors, and varies greatly depending on the situation and individual skill. In high-level poker games, players try to spot "tells" from other players that betray a bluff. In fact, players tend to become more rigid when they bluff, reducing the movement of their head in particular [Mandjes, 2019]. But how well can people spot tells in other scenarios?

In text, there is no movement data, so readers cannot use rigidity as a guide. Indeed, the lack of physiological information generally means that most people don't reach a confident conclusion from reading. There is also a bias toward truth: when lacking a reason to doubt,

people generally will give the writer the benefit of the doubt and assume that what they are reading is true [Levine, 2014]. Even when subjects are given the knowledge of the certain existence of deceptive text, they are generally not good at separating it from truthful text [Barsever et al., est 2022]. What exactly humans use as guideposts to make their determinations is also a mystery, one that I intend to look into.

What makes this task so different from a sentiment classification? Deception is fundamentally different to all other forms of text-based classification because high-quality deception is, by its very definition, difficult to spot. With sentiment, the author has a certain intent and message they want to convey, and their goal is that the reader should identify that message as clearly as possible. But with deception, the ideal scenario for a motivated deceiver is that the reader never notice the actual intent of the author. This places a conflict between the reader and the author that is not present in any other text classification task, and increases the difficulty in doing so. Humans, in fact are extremely poor even at identifying if a review is generated by a human or an artificial intelligence [Hovy, 2016]. So if humans are not good detectors of deception, perhaps we can turn to a more algorithmic judge.

1.4 Natural Language Processing

The ability of computers to understand text or spoken words is referred to as Natural Language Processing, or NLP. NLP covers a dizzying range of tasks, including conversational agents, spellchecking, machine translation, web-based question answering, and speech recognition. These are difficult tasks for computers because they require certain knowledge about language in order to perform.

In order for a word-counting software to perform, it must first know which bytes of data and groups of symbols constitute a word. Similarly, while it is intuitive to think that a

question-answering program needs knowledge of certain facts, on a more basic level it needs knowledge of grammar in order to correctly parse the question. "I had my car cleaned" and "I had cleaned my car" contain the same words, but have different meanings because the structure is different. We make use of this kind of algorithm near-constantly, as it is the basis for services like web-based search engines. When someone searches for "What year did J.R.R Tolkien write The Hobbit," that search engine needs to know what the subject of the question is (the year the book was written) without confusing it with the clarifying details in the question (The answer should not be a summary of the Hobbit). The framework that a system uses to parse the language it is given and apply the "knowledge" it possess is often called a language model.

1.5 Language Models

Working with natural language processing generally requires two things: a corpus of data and a model to interpret it. Language models come in a huge variety of forms, from a simple unigram to a highly complex neural network. When you perform speech recognition [Kuhn and De Mori, 1990], machine translation [Vaswani et al., 2017], part-of-speech tagging [Cutting et al., 1992], handwriting recognition [Zamora-Martinez et al., 2014], or any of a dozen other natural language tasks, you are using a language model. At its most basic, a language model is a probability distribution over a given set of words. This forms the basis of any tool constructed to work with text. A classifier finds the probability distribution of its classes from a given input of words. A generator finds the most likely word from its vocabulary given some context.

Where these language models often get put to use is in the area of machine learning. Machine learning algorithms have been used to work with text in a variety of different tasks. Support Vector Machines (SVM) are some of the most popular algorithms [Noble, 2006]. A SVM

can be given a dataset and then trained to draw boundaries between the features of the data, allowing it to assign labels. Vogler and Pearl used a SVM to classify the Ott corpus according to linguistically defined features, demonstrating their viability on this task [Vogler and Pearl, 2019]. The other common model structures are neural networks. Neural networks are powerful and versatile, able to classify texts by sentiment [Tang et al., 2015], identify the author of a text [Shrestha et al., 2017], or generate text of its own [Sutskever et al., 2011]. Unlike a SVM, a neural network algorithm is based on the interrelation of layers of artificial neurons. Both neural nets and SVMs are effective classifiers, but each has their own separate advantages. SVMs are very efficient with small batches of data, but do not scale as well as neural networks as the size and dimensionality of the data increases.

1.6 Neural Networks

Neural networks, which this research makes extensive use of, are powerful computational tools. They consist of some very basic components that can be woven together into sometimes hugely complex algorithms. The elemental unit of a network is the neuron. These neurons are designed to replicate biological neurons, in that they receive some sort of input signal, perform a given operation, and then send out an output signal. When multiple neurons are linked together, such that some neurons inputs come from the outputs of one or more other neurons, that creates a network. What allows the network to learn is the ability to modify the connections between the neurons (called the synapses), which changes how the network calculates from a given input. This is usually done through a loss function that determines the distance between the network's output and the desired output. Depending on how the network is configured, many different models can be constructed to fit a given task.

1.6.1 The Transformer

One type of network that is of particular interest is known as a transformer network. This network was designed by Vaswani et al [Vaswani et al., 2017] for use in machine translation. What makes the transformer different from other types of networks is the use of "attention." Attention does not require any recurrent network units, in fact their function is surprisingly basic for a neural network: only weighted sums and activations are necessary. The architecture is shown in Figure 1.1, and consists of two main blocks, the encoder and the decoder. On the left is the encoder, and the on the right is the decoder. Both blocks contain N repetitions of an attention and feed-forward network.

The self-attention mechanism that gives the transformer its power has three main elements: the query, the value, and the key. Each input vector is compared to the other input vectors obtain its own output (the query), the j-th output (the key) and the the output vector after the weights are established (the value). Each vector is calculated from the input using a different learnable weight layer. Since each weight matrix comes from the same input, it is possible to apply the attention mechanism of the input vector to itself, ergo "self-attention." Computing the dot product of the query and all of the keys, and then applying a softmax function to normalize the result, gives a weight matrix we can apply to the value matrix. When we do, the resultant vector shows which parts of the input, usually certain words, are important to focus on and which can be ignored; in other words, which part of the input to pay *attention* to.

This computation creates one "head" of attention, which focuses on an entire sentence at once. What gives the transformer its flexibility, its ability to understand contextual information, is applying this computation to multiple heads that each focus on a chunk of the sentence, then concatenating the results. Adding a positional encoding to the input allows the network to know which parts of the input occur in what order.

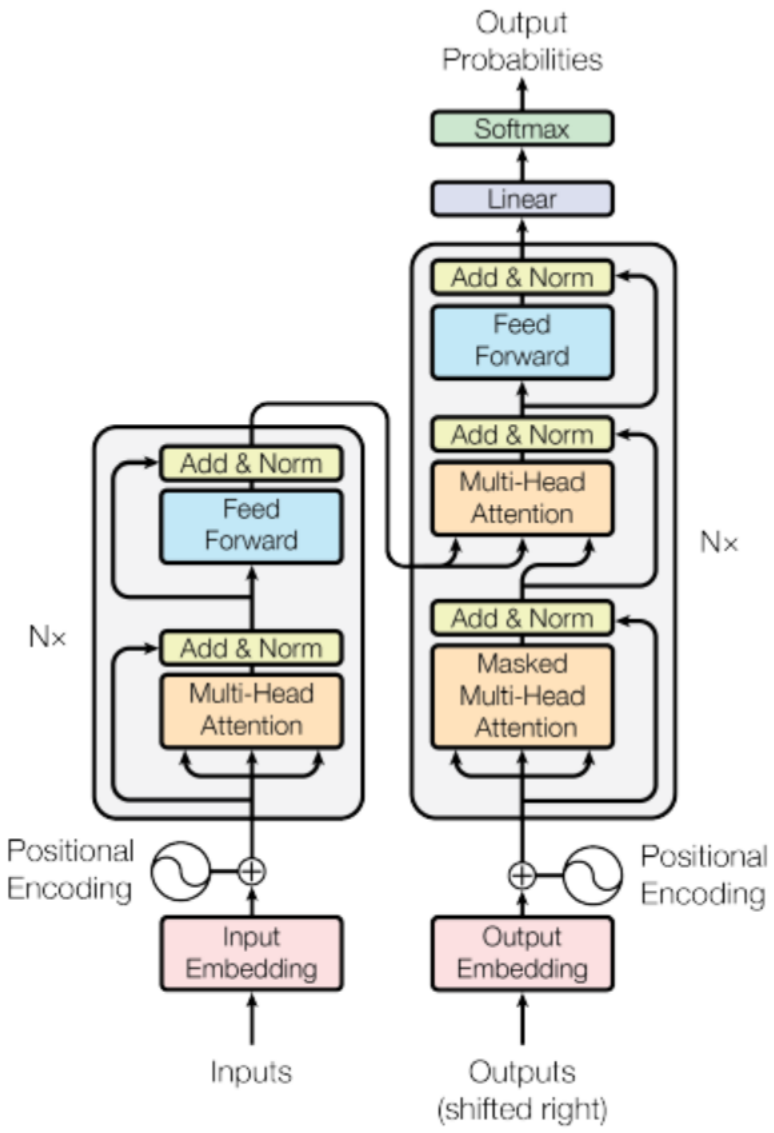

Figure 1.1: Transformer network architecture. From "Attention is all you need" paper by Vaswani et al., 2017 [Vaswani et al., 2017]

The encoder then is the amalgamation of these elements, applying a positional encoding to some input sentence and passing that input through N layers of attention, where the output of the last layer becomes the encoded sequence. This sequence becomes the input for the decoder, which uses most of the same components as the encoder. Unlike the encoder, each decoder unit's first layer is so-called "masked" attention layer, which prevents a position from attending to future positions, such that their predictions only depend on the known

outputs at positions prior to itself. The final output of the decoder is passed through a large linear layer and normalized to obtain the logits, which contain the probabilities for a certain word at a given position. The encoder and decoder combine to form the transformer in total. This process allows, for example, a sequence to be presented in English, encoded into a sequence that contains the meaning of the words as they relate to the other words in the sequence, and then decoded into a sequence in another language but with an equivalent meaning.

One model that has taken this architecture and applied it to great use is BERT (Bidirectional Encoder Representations from Transformers). Devlin et al built on the work of Vaswani et al to create the structure known as BERT, using many instances of self-attention networks to learn contextual representations of text. What makes BERT so much of an improvement on previous networks is its bidirectionality. As mentioned, the Vaswani transformer decoder can only attend to a sequence in a single direction; inputs further in the future of a position do not factor into the coding for that position. This means that every token can only attend to the the context to its left. BERT however boasts bidirectionality, both previous and future tokens can influence the meaning of an attended token. BERT is otherwise extremely similar in makeup to the original transformer.

BERT's unique training method that allows it to have meaningful bidirectional context is its masked language modeling. Normally, trying to train bidirectionally is counterproductive, since sweeping a sequence from both left to right and right to left will allow a token to "see," and therefore get context from, itself, rendering the task both trivial and unhelpful. BERT's training method replaces a random selection of words with special [MASK] tokens, which do not give information about themselves. BERT then tries to predict the true identity of the masked word given the context of the surrounding words. This allows BERT to train deep bidirectional encodings of its tokens.

BERT's other flagship task is called Next Sentence Prediction. Next Sentence Prediction

is similar to question answering in that it deals with the relationship between separate sentences, but instead of trying to generate an answer to a random question this model predicts whether a given sequence B is the sentence that follows, or is the 'next sentence,' of sentence A. This is ultimately a simple binary classification task, but the training for this task results in some useful transfer learning that is beneficial to other tasks like question answering. The structure of this model means that all BERT input is in truth an input in three parts. First is the token embeddings, the sentence is tokenized into component parts, such as words, punctuation, suffixes and prefixes of words, and certain proprietary tokens like the classifier ([CLS], and separator ([SEP]) tokens. Second are what is called 'segment embeddings,' which are used solely in the next sentence prediction task. This embedding tells BERT which tokens belong to sequence A and which belong to sequence B. The final embedding is the positional encoding, which tells BERT, as in Vaswani's transformer, the order of the tokens in the sequence. A visual representation of this input is shown in Figure 1.2.

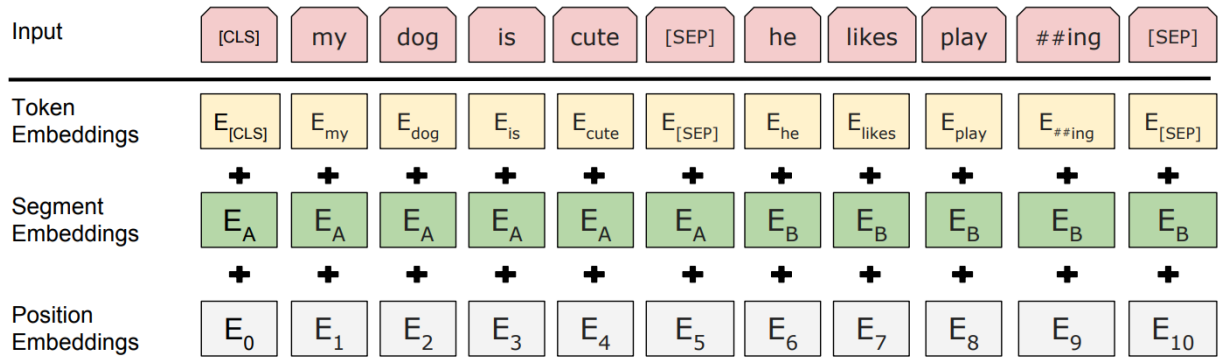

Figure 1.2: BERT input breakdown. For each token BERT encodes the ID of the token (token embeddings), which sentence it is part of (segment embeddings), and where it is in the sequence (position embeddings). From "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" paper by Devlin et al., 2018 [Devlin et al., 2018]

The deep bidirectional representations BERT learns allows it to excel in a broad array of natural language tasks. BERT has proven highly competitive in multiple areas including sentiment classification and next sentence prediction [Devlin et al., 2018]. Barsever et al

[Barsever et al., 2020] were able to utilize BERT in order to perform deception detection on the Ott dataset, proving its viability and power in that arena and setting the state of the art. Their model was able to achieve an accuracy of 93.6% on the Ott Deceptive Opinion Spam corpus. They also used BERT as a generative model to produce machine-created samples of both truth and deception, and identified some linguistics trends such as deceptive samples being less varied in the parts of speech that they used.

1.7 This Thesis

In the following chapters, I use neural network language models, including BERT, to unravel the problem of deception detection in text from three angles. First, in Chapter 2, I apply BERT-based language models to existing deceptive text data. I then examine these models' behavior to determine what aspects of the data allow the model to make its determination. Then, in Chapter 3, I create an entirely new corpus of deceptive text data, one that uses gaming techniques to produce high-quality samples in large quantities. Finally, in Chapter 3, I deploy this new dataset along with existing data to Amazon Mechanical Turk in order to obtain mass judgements on the data from human subjects, which allows us to analyze what drives human judgement of deceptive text, regardless of whether that judgement is correct or not.

More specifically, Chapter 2 is a work that describes several forays into analyzing deceptive text using neural networks. My goal with this work was to establish certain basic principles that have to do with the validity of this entire thesis. This work begins with the construction of a new state-of-the-art classifier. That classifier is formulated from a modified version of BERT, and is a valuable tool in its own right. Its true value comes not in the ability to use it on a piece of text though, but from using that accuracy to analyze its behavior on deceptive text. I utilize that classifier by first performing two ablation studies. The first study is on

the word level, and is meant to see whether certain parts of speech are more influential to the classification. The second is sentence-level, in order to assemble a group of sentences that produce a strong response from the classifier. Following those findings, I built a new network based on the new classifier, a BERT-based Generative Adversarial Network. With this new tool, I could create new sample of deceptive or truthful text out of whole cloth, and compare them to the samples drawn from the Ott Deceptive Opinion Spam dataset. This chapter is based on a paper that was published in the 2020 International Joint Conference on Neural Networks (IJCNN).

In Chapter 3, I present a project that contributes a new corpus of text for use by researchers all throughout the field. I call this new corpus the Motivated Deception Corpus, or the Two Truths and a Lie corpus. This corpus is born of a simple problem I found with publicly available corpora: there was no indication that the liar who created the text was trying their best. In the real world, people who create deceptive text do so with the intent to be believed, and so they have a vested interest in not getting caught. That attitude, and thus effort, is difficult to simulate with traditional data-gathering methods. To account for this, I gameified the process. By basing my gathering on the party game Two Truths and a Lie, I added something crucial to the process: a reason to succeed. When you can win and get more money by making convincing deception, you put more effort in, just like a real-world deceiver. I believe that this corpus, and the techniques I used to create it, will be a valuable tool for those looking to test their models on more realistic data. As an added bonus, I also gathered behavioral data in the form of their keystrokes, increasing the depth of the data available. This chapter is based on a work currently being considered for publication at Behavioral Research Methods.

In Chapter 4, I demonstrate a way to use my newly-created corpus by performing a study on human judgement. Since I now had a corpus of realistic data, I wanted to see how people would respond to it. I created a study on Amazon's Mechanical Turk and challenged the

Turkers therein to undertake the same task as the players who created the corpus: spot the one deceptive story amid the two truthful ones. This allowed me to collect multiple judgements per sample while still incentivizing people to search as hard as they could, since they would be paid for every lie they spotted. I created a new iteration of the Bayesian Cultural Consensus model to analyze these responses. By feeding the responses of the Turkers into this model, I could extract which answers were *perceived* as truthful or deceptive by humans, regardless of whether they were truly deceptive or truthful. By analyzing these perceived truthful and deceptive stories, I gleaned some trends on what humans view as deceptive, such as the presence of extreme or negative words, or the structure of a story as a narrative or a factual presentation.

Chapter 2

Build a Better Lie Detector

2.1 Introduction

To understand deceptive text, I need to establish some baselines. First, I need to show something seemingly trivial: that there is a difference between truthful and deceptive text. It's a natural question to ask, while it is generally accepted that there is a perceivable difference between a verbal truth or lie, some might say that there is no way to tell the difference in text. To answer this question, I apply a tool that requires no human judgement: a neural network.

The job of a neural network is to take inputs and learn how to classify them. The more input cycles you feed it, the more it can modify itself and determine how to classify inputs beyond its training set. By feeding it the current gold-standard Ott Deceptive Opinion Spam dataset, I were able to see if a neural network could learn to differentiate between the truthful and deceptive samples. If it could achieve a high classification accuracy, that is a solid indicator that there are differences to be exploited between the two types of text.

To achieve this goal, I built a state-of-the-art classifier that can learn the patterns that constitute a deceptive review, and then analyzed that classifier to identify those patterns. To this end, I constructed a machine learning tool utilizing BERT. BERT (Bidirectional Encoder Representations from Transformers) is a recently developed neural network architecture that is pretrained on millions of words and is capable of forming different representations of text based on context [Devlin et al., 2018]. By applying BERT to deception detection, I can use it to form a powerful classifier of deceptive text. After that, extracting the rules that BERT forms to classify the text can help us understand what patterns underlie deceptive text.

Our BERT-based classifier proved to be a useful tool for this study, defeating the state of the art on the Ott Deceptive Opinion Spam corpus and facilitating analysis on how it determines deceptive from truthful text. The rules it generates are still not completely clear, but my ablation study, where each part of speech (verbs, nouns, etc) is removed and the network's performance is monitored, has indicated that certain parts of speech such as singular nouns are more informative than others, as their removal resulted in the sharpest drop in accuracy.

I also performed part-of-speech analysis on 'swing' sentences—sentences shown to be informative to BERT's decision making. My findings indicate that truthful sentences have more variance in what parts of speech occur. This provides evidence that there is a commonality in the structure of deceptive text that is less present in truthful text. This evidence is reinforced by the Generative Adversarial Network that I created, where a text generator based on BERT must try to create samples that can fool the BERT classifier into thinking they are real examples. The samples produced by my generator are easily recognized by the classifier as truthful or deceptive and reproduce many of the same trends seen in the swing sentences, particularly that many parts of speech appear with less variation across samples. This again points to deceptive text being more formulaic and less varied than truthful text.

2.2 Related Work

Ott et al. [2011] developed the Ott Deceptive Opinion Spam corpus, which consists of 800 true reviews from TripAdvisor and 800 deceptive reviews sourced from Amazon Mechanical Turk. He used this corpus to train Naïve Bayes and Support Vector Machine (SVM) classifiers, achieving a maximum accuracy of 89.8% with an SVM utilizing Linguistic Inquiry and Word Count (LIWC) combined with bigrams. The Ott corpus is one of the most commonly used gold-standard corpora in deception detection tasks. Other, less widespread corpora include the LIAR fake news dataset [Wang, 2017], Yelp dataset in Feng et al. [2012], and the Mafiascum dataset [de Ruiter and Kachergis, 2018].

Vogler and Pearl [2019] used a support vector machine operating on linguistically defined features to classify the Ott corpus. They were able to achieve an accuracy of 87% using this method. Xu and Zhao [2012] train a maximum entropy model on the Ott corpus and were able to achieve 91.6% accuracy. Li et al. [2014] tried to find a general rule for identifying deceptive opinion spam using features like part-of-speech on several datasets including the Ott corpus, achieving 81.8% accuracy on Ott. [Ren and Ji, 2017] expand on this work by using a recurrent neural network on the same data, improving the accuracy to 85.7%.

Hu [2019] used a variety of models to identify concealed information in text and verbal speech, best among them a deep learning model based off bidirectional LSTMs. Concealed information, in this context, refers to when a person has knowledge about a subject and is withholding it, as compared to Hu's definition of deception where someone fakes knowledge they do not have. Hu created a corpus of wine tasters evaluating wines and encoding in various ways such as n-grams, LIWC, and GloVe embeddings [Pennington et al., 2014] based on the recordings. The LSTM model using these features achieved an f-score in identifying the presence of concealed information of 71.51, defeating the human performance of 56.28.

Jin et al. [2019] put BERT's robustness to the test by attacking its input in text classification

and textual entailment tasks. They did so by calculating an Importance score for each word in an input sequence, and then perturbing that input by substituting semantically similar words to replace the most important words. Using this method they produced input that was classified correctly by humans but was overall nonsense to BERT. Similarly, Niven and Kao [2019] attempt to examine what is informative to BERT in the Argument Reason Comprehension Task, where BERT must pick the correct warrant to follow a claim and a reason. They found some words, such as the word 'not' acted as a statistical cue that signaled it as an answer. Removing these words dropped BERT's accuracy dramatically.

Wang and Cho [2019] demonstrate BERT's viability as a generative model by utilizing its ability to predict masked words. BERT faces challenges as a traditional language model because it is bidirectional and depends of the left and right context of a word in order to predict it. Wang and Cho circumvent this problem by providing BERT with a full sequence of masked tokens and predicting each one in a random order until the full sequence is unmasked. This method also allows BERT to receive noisy inputs by setting some of the masked tokens to random tokens. Using BERT in this manner generated more diverse sequences than OpenAI Generative Pre-Training Transformer [Radford et al., 2018], with the tradeoff of somewhat higher perplexity.

2.3 Methods

2.3.1 Classification

The network I use for this work is based on BERT. I experimented with several configurations and modifications of BERT, and the highest performing network was the base BERT model with a few additional layers on top. I added a bidirectional LSTM, attention layer, and dense linear layer on top of BERT as a classifier (see the blue components of Figure 2.1).

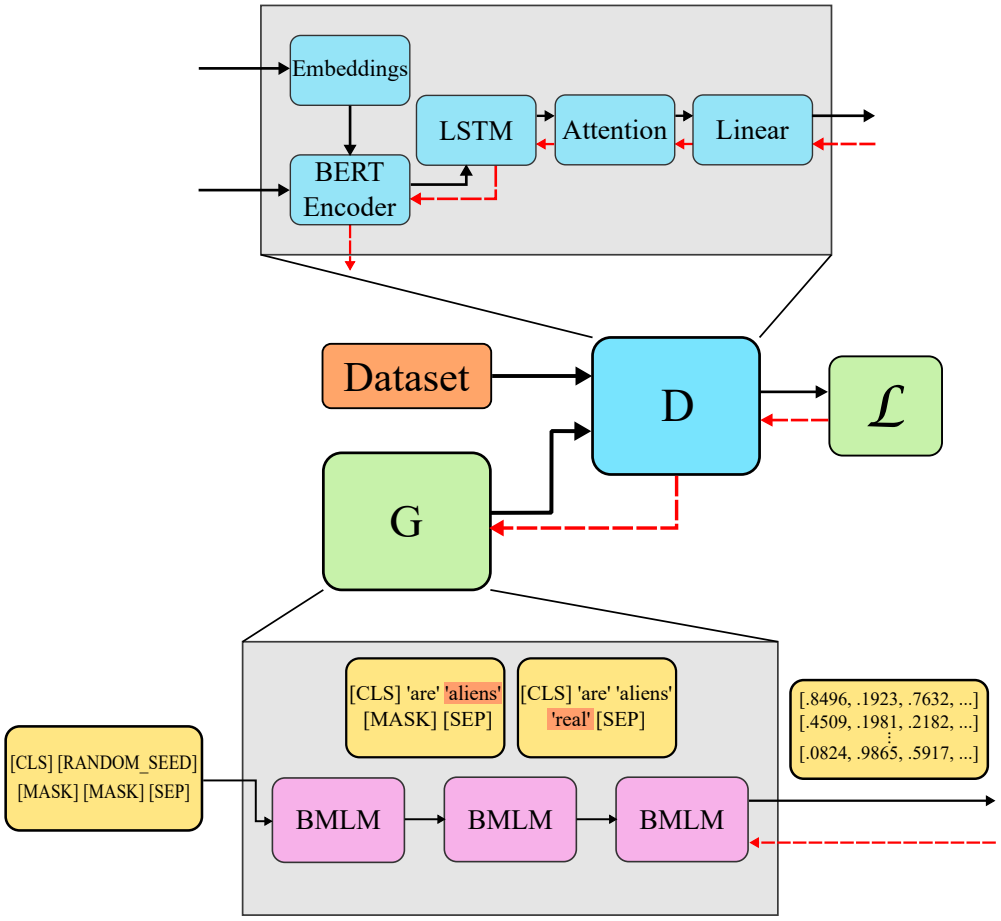

Figure 2.1: A block diagram of my BERT-based Generative Adversarial Network. The discriminator (D) is composed of a BERT embedding layer, a BERT encoder, a directional LSTM layer, a self-attention layer, and a dense linear layer that produces the final classification. The embedding layer converts integer input (which represent tokens in BERT's vocabulary) and converts them to a set of 768 length float vectors. The encoder converts those vectors into a single 768 length vector that encodes the entire sequence. Samples drawn from a dataset are converted to integers and presented to the embedding layer, while generated samples, being already float vectors, are presented directly to the encoder. When training the generator (G), a sequence of [MASK] tokens plus a random seed token is first presented to a model of BERT for Masked Language Modeling (BMLM). The generator selects a token at random and predicts it, replacing the [MASK] token with its prediction. This process is repeated until all tokens are predicted, producing a full sequence which is then converted to float vectors. This tensor passes through the BERT encoder and up the discriminator. After the loss is calculated, it is backpropagated through the discriminator and the BMLM (shown by the red arrows). Because the generator must cast integers to form the intermediate sentences, only the last instance of the BMLM is differentiable and can be backpropagated, however as all the instances of BMLM share parameters this is enough to train the generator.

BERT has several advantages over previous methods. First, BERT performs well in a wide variety of contextually sensitive language tasks due to being able to detect when the meaning of a sequence has changed depending on context, allowing it to detect subtle differences in phrasing [Devlin et al., 2018]. This is thanks to its bidirectional masked language model training, which allows it to attend to both past and future tokens when observing a given word. BERT also requires significantly less preprocessing of data than previous methods. Inputting language into a neural network requires the researcher to convert the tokens to numbers that the network can understand, usually through an embedder like GloVe [Pennington et al., 2014]. BERT has a native tokenizer and embedder that reduces the complexity of this step.

The primary idea behind most prior work is to extract predefined features (such as bigrams or part-of-speech counts) from a sample and classify according to those features. This is fine for testing specific hypotheses, but for this task the idea is simply to differentiate the two classes by any means possible. BERT requires no predefining of features and is free to develop its own rules. The BERT model I used is the publicly available `bert-base-uncased` pretrained BERT model for PyTorch from huggingface's transformers library[1].

I used the Ott Deceptive Opinion Spam corpus to benchmark the network and compare it to previous approaches. The task is a simple binary classification, with the network classifying any given story as either a truth or a lie. 80% of the reviews form the train set, which will be used to train the network. The remaining 20% become the test set, used to evaluate the network. In each training epoch, the training set is presented to the network in random batches of 8 until the entire set has been presented. Training lasted for 100 total epochs.

[1] https://github.com/huggingface/transformers

2.3.2 Part-of-Speech Ablation

After establishing the accuracy of the classifier, I began my first investigation into which parts of the input are the most important. The best way to do that was by performing an ablation study on the network after training. First I split each review in the test set into its individual tokens. Then, I tagged each token of each review in the test set with its part of speech using the Natural Language ToolKit's part-of-speech tagger[Loper and Bird, 2002]. After identifying each token by its part-of-speech, I went through each one and replaced tokens that corresponding to the part-of-speech of interest with a [MASK] token. This token tells the BERT classifier that there is a token there, but it must infer what it is. I then ran these ablated inputs through the classifier and observed the resultant drop in accuracy. If BERT had a larger-than-normal drop in accuracy, that type of word could be judged to be important to the original classification. This ablation was done 10 times for each part of speech, each with a freshly trained classifier.

2.3.3 Identifying Swing Sentences

After testing for certain parts of speech, I tested the same idea on the sentence. In an alternative route to identifying informative parts of the input, I identified certain "swing" sentences that BERT considered highly informative for classification. To identify these sentences, I started with the trained classifier, just as in the part-of-speech ablation. Then, for each review in the test set, I constructed a new input. This new input would be exactly the same as the old one, with one crucial difference. One of its sentences would have all of its tokens replaced with [MASK] tokens. This process would be repeated for each sentences in the paragraph until I had one paragraph for each ablated sentence. One-sentence entries are excluded from this process due to being rendered completely inert by the process. An example is shown below.

In this case, I am not watching for an overall drop in accuracy, but a case-by-case altering of the classification. If the base paragraph had previously been correctly classified, but removing a sentence from that paragraph causes BERT to switch classifications and label it incorrectly, that sentence was marked as one that was important to the original classification and therefore an exemplar of deception or truthfulness. These sentences become the 'swing' sentences, sentences that cause the classification to swing one way or the other. I then analyzed those sentences' parts-of-speech to see if there are any observable differences between them. I track the mean amount of times a given part of speech appears per sample where they are used, the standard deviation of the same, and percentage of samples where they appear at least once.

Before

[CLS] We stayed for two nights for a meeting. [SEP] It is an upscale chain hotel and was very clean. [SEP] The service was very good, as the hotel front desk employees were kind and knowledgeable. [SEP] The rooms are decent sized and have soft mattresses. [SEP] The restaurant has good seafood, but was a bit expensive. [SEP] We would come back again. [SEP]

After

[CLS] we stayed for two nights for a meeting . [SEP] it is an upscale chain hotel and was very clean . [SEP] the service was very good , as the hotel front desk employees were kind and knowledgeable . [SEP] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [MASK] [SEP] the restaurant has good seafood , but was a bit expensive . [SEP] we would come back again . [SEP]

It is worth asking why I am doing this analysis only on these swing sentences rather than on the whole dataset. The answer is simple: not every part of the text in the dataset is informative. It is inevitable that some parts of the input are essentially noise to BERT, not providing evidence either way. With this ablation, I am aiming to identify what BERT is focusing on when making its classifications. By limiting the analysis to these swing sentences, I narrow the domain to what BERT considers to be strong examples of truth or deception. Examining them will then provide the most concentrated evidence of the differences between truth and deception.

2.3.4 GROUCH

The final method proposed to expose the patterns in truth and deception is a Generative Adversarial Network based on BERT, shown in Figure 2.1. I call this new network the Generator of Realistic Obfuscation Under Combative Honesty, or GROUCH. The ability of BERT to function as a generative model is vital if one wants to use BERT in a Generative Adversarial Network (GAN). Goodfellow et al. [2014] created the GAN to be a unique network system that would allow a network to generate plausible samples by getting feedback from a discriminator network, usually a reliable classifier. First, a *generator* creates a data sample from a random 'latent' input. This generated sample is intended to imitate a sample from a real dataset as closely as possible. This sample is then presented alongside samples from the real dataset to the *discriminator*. The discriminator then attempts to decide if the input it is receiving are from the real dataset or from the generator. The loss from the decision gets propagated back through both the discriminator and generator. They both learn at the same time; the discriminator learns to tell the difference between fake and real, and the generator learns how to exploit the discriminator and make convincing samples. A truly successful generator, assuming the discriminator is high quality, should keep the discriminator at around 50% accuracy, at the point where it has to guess randomly which

samples are real or fake.

In this setup, I declare the BERT model I have been using as a classifier as the discriminator. For the generator I use the BERT-based implementation by Wang and Cho [2019], which takes advantage of BERT's ability to predict masked tokens to act as a generative model. By generating samples from latent variables composed of mask tokens and having those samples evaluated by the discriminator, the generator learns to create samples that can fool the discriminator into thinking that the sample came from a real dataset. This way, both the discriminator and generator utilize BERT.

The generator network exploits BERT's masked language model abilities. One of BERT's basic functions is the ability to predict the true identity of a masked word given its surrounding context [Devlin et al., 2018]. I expand on the work of Wang and Cho [2019] to allow BERT to produce entire sequences from scratch. First, an entirely masked sequence is presented to the generator, as well as a random seed token at the beginning to provide noise. The generator then selects a random token and tries to predict it, producing a probability distribution of the tokens that it could be, which is then sampled to provide its prediction. This new sequence is fed back into the generator, where a different random token is selected and predicted. This continues until all the tokens have been predicted, forming an entire sequence. A side effect of this iterative process is that the generator must sample its output to form integers to represent the intermediate sentences. This means that only the last instance of the masked language model is differentiable. However, since all the parameters are shared across instances, this does not noticeably harm the generator. The generator produces a sequence of 48 tokens before transforming it to a 50 token sequence by prepending a [CLS] token, which allows the discriminator to classify the sample, and appending a [SEP] token, which signals to BERT the end of a sentence. I then perform part-of-speech analysis on the samples that successfully fool the discriminator into believing that they are real samples, if any are produced.

Source	Accuracy
Ott et al. [2011]	89.8%
Vogler and Pearl [2019]	87.0%
Xu and Zhao [2012]	91.6%
Ren and Ji [2017]	85.7%
BERT	**93.6%**

Table 2.1: Comparison of accuracies on the Ott corpus.

I perform two runs of this GAN: once each for deceptive and truthful sentences. I use the Ott corpus to provide the real-world examples of both. This allows the BERT generator to generate its own examples to mimic what is truthful and what is deceptive. This will allow the generator to exploit the features that the discriminator is using to identify truthful and deceptive sentences. The advantage of this approach is that the generated sequences do not need to fool a human expert or even produce recognizable English; they just have to exploit the rules that BERT creates, which should shed some light on what those rules are. In fact, it is almost preferable to produce nonsense as it reduces the likelihood of a narrative to distract from the underlying pattern. I perform part-of-speech analysis on the generated truthful and deceptive sentences to analyze the representational similarity between the two cases.

In addition to this, I also make a new sub-dataset derived solely from the generator. Since I can run the GROUCH network to generate both deceptive and truthful samples, I can use it to formulate a new balanced corpus in imitation of the Deceptive Opinion Spam dataset. I present the generated mini-corpus created by GROUCH to the original BERT classifier as a classic two-label discrimination task. The BERT classifier will be trained solely on the Ott Deceptive Opinion Spam corpus. The question then is, will the Ott-trained discriminator be able to classify the generated samples as well as the original Deceptive Opinion Spam samples? Note that in this case I am not testing its ability to discriminate real samples from generated ones this time; I want to know if the differences between truth and deception in

the generated samples can be picked up on by a discriminator that has never seen a sample from the generated dataset before.

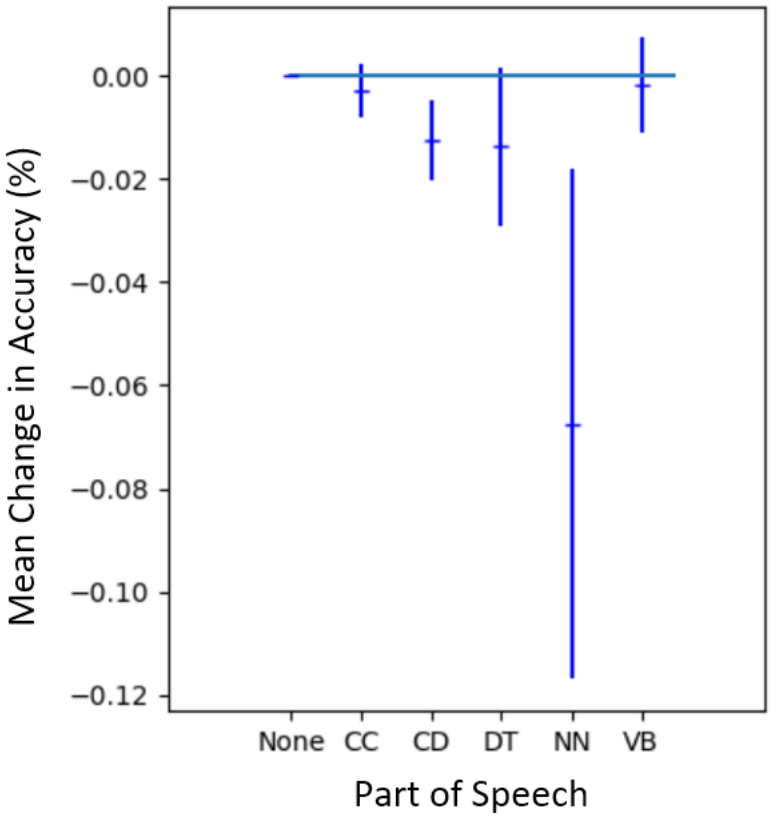

Figure 2.2: The mean results of the ablation study over 10 runs. The error bars are the standard deviation. The removed parts of speech shown here are None Removed, Coordinating Conjunction, Cardinal Digit, Determiner, Singular Noun, and Verb.

2.4 Results

2.4.1 Classification

BERT reached an accuracy of 93.6% (table 1), an improvement of 2% over the next best method, beating the state of the art in deception detection on the Ott dataset. This proves the network's efficacy as a tool in evaluation deceptive text. This jump in accuracy is significant since, unlike other methods which have the conditions and factors of interest

baked into the model, BERT must learn its rules and features unsupervised. That allows BERT to find the best solution unrestricted by preconceived rules, and therefore attain the best accuracy. BERT has achieved the first step for this work: being able to accurately classify deceptive text, allowing me to investigate the methods it uses to do so.

2.4.2 Ablation

The ablation study (Figure 2.2) revealed that the network is insensitive to most parts of speech being removed, although some have a slightly stronger impact. One in particular seems to be responsible for a larger than normal reduction in accuracy. When the singular nouns (NN) were removed, the network accuracy dropped by 2 to 12 percent. This is close to triple the effect of the next-highest impact, the determiners (DT). This may indicate that singular nouns are stronger indicators of deception or truth than any other part-of-speech; however given the prevalence of singular nouns in everyday language it is possible that removing them makes the review less comprehensible overall and harder to classify.

2.4.3 Swing Sentences

BERT identified 69 truthful swing sentences and 148 deceptive swing sentences. Examples from both classes are shown in Tables 2.2 and 2.3. The results of the part-of-speech analysis and percentage occurrence are shown in Figures 2.3 and 2.4. Many parts of speech occur less frequently in deceptive sentences than in truthful sentences, and the standard deviations tend to be much lower. Those same parts of speech also appear (at least once) in a higher percentage of samples for truthful sentences than deceptive sentences. It is possible that truthful sentences tend to have more varied parts-of-speech, and as a result are less consistent in which parts of speech are used. Deceptive sentences, meanwhile, seem to draw from a shallower pool and have less variation. This indicates that the deceptive sentences are more

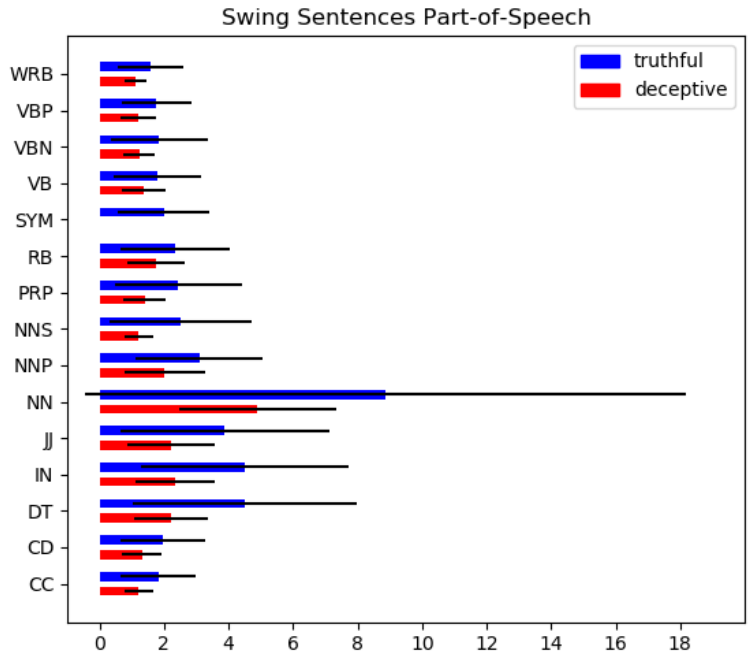

Figure 2.3: The part-of-speech analysis of the swing sentences. Bar length indicates average number of occurrences per sentence with error bars representing standard deviation.

formulaic and follow a more consistent structure than the truthful sentences. This in turn suggests that truthful text may not have a specific 'marker' designating it as truthful, but that marker might exist for deceptive text.

Truthful Swing Sentences
recently returned from a two - night stay at ambassador east hotel.
the recently remodeled affina was amazing - from the 6 choice pillow menu to the stocked " pantry and refrigerator " to the brand name bathroom ammenities.
once in a while you come across hotels with great service, well thought out rooms in a perfect location (right off magnificent mile).

we could easily walk to the red line or access the bus lines along michigan avenue.
a bunch of us got together and we had a great time in this hotel we asked for limes and they gave us like a punch bowl of them the rooms were so awesome you really have to see it to believe how extradionary this hotel is i love the decorations on every floor and being surrounded by such elegance.
as a royal ambassador member, they upgraded me to a beautiful junior suite with a separate living and working area and 2 bathrooms!
ideal position, lovely quiet rooms, good facilities, complimentary breakfast well received and the manager's evening drinks reception excellent ; we always tipped the staff who were serving our drinks.
as we were diamond members we got upgraded to a suite which was great because we were provided access to their executive lounge (free food & drinks).
beautifully appointed, professionally staffed, comfortable, well - located, and elegant, with wireless internet access (plus free access in an alcove on the second floor), fitness center (a little small but fine if you use it at the right time), good energy clientele, easy access to michigan avenue, next door to 24 - hour restaurant, 2 blocks from 2 starbucks, elevators that are there before you get your finger off the button - they thought of everything.

there are other good choices in town, but nobody has what the james offers : superior service - aka class meets efficiency -, metropolitan design and comfort with an attention to details i found only at the w in seoul, and what always plays a role meaning a convenient location... then they get to you with small unexpected touches that make the difference ie : i was about to take off and drive north, it was a hot july day... one of the staff members handed me a bottle of water.

Table 2.2: Ten truthful swing sentences as identified by our model, reconstructed from the tokenizer. The full list is available in the Appendix.

Deceptive Swing Sentences
i loved staying at the hard rock hotel in chicago, not only is it an amazingly friendly atmosphere, but they give me the option to bring my pet with me.
the ambassador east hotel in chicago is a fantastic up - scale hotel to stay in while visiting the windy city.
the affinia chicago is a wonderful place to stay, my husband and i stayed there for a week to visit some family and had an amazing time.
the hard rock hotel chicago is great alternative to ordinary hotels.
upon arrival at the ambassador east hotel in chicago, i was immediately impressed with the courtesy and attentiveness of the staff.

the magnificent mile in chicago is a great place to visit, and staying at the affinia chicago just made it that much better!
elegant and luxurious with a beautiful ocean view.
my husband and i went over the holidays to see my family and we stayed at this hotel.
amalfi hotel chicago has several factors that make it one of the best hotels in the chicago area and an experience you will not forget in a long time.
the palmer house hilton in chicago is by far the best experience i have ever had away from home.

Table 2.3: Ten deceptive swing sentences as identified by our model, reconstructed from the tokenizer. The full list is available in the Appendix

2.4.4 GROUCH

The BERT-based generative network was able to produce samples of text that were easily identifiable as truthful or deceptive to the classifier, if not to a human. There is a sharp drop in coherency in both truthful and deceptive text after training compared to before it is trained. Fortunately, readability of these samples is not necessary for them to be useful. The neural network after, does not truly understand language, and is only looking at the presence or absence of certain features. When eighty generated samples were presented to the trained classifier from earlier, the classifier was able to identify all of them with 100% accuracy, even though it was only trained on the Ott data and never trained on generated samples. This indicates that the generated samples show strong resemblance to what BERT considers either ideal truth or ideal deception. GROUCH was therefore able to mimic the

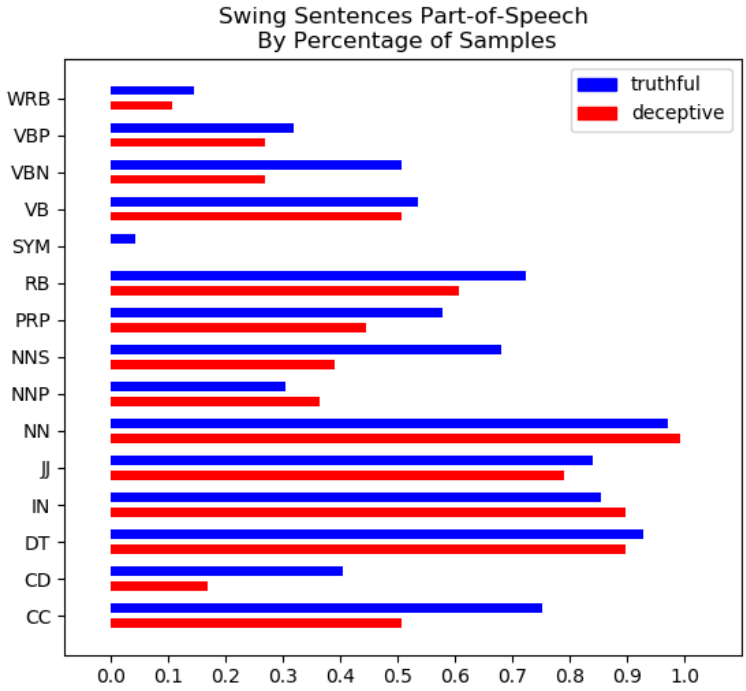

Figure 2.4: The parts of speech by the percentage of samples they appear in at least once for swing sentences.

essence of the samples in the Deceptive Opinion Spam dataset, even if its own samples were totally incomprehensible.

Samples of truthful and deceptive sequences that successfully fooled the discriminator are shown in the boxes below, as well as a sentence produced before training for comparison. The sentences were produced in all lowercase, with the [CLS] and [SEP] tokens added after the fact to fit BERT's input rules.

Untrained Generated Sequence

[CLS] greyhound trains were running on behalf of the university , and shaw interested in improving access to the food markets and in the improvement of healthcare . the hospital was put under much pressure by the government , also underperformed at parliament in that year . [SEP]

Figures 2.5 and 2.6 show the results of the part-of-speech analysis on the generated sentences. Some of the same trends that are visible in the swing sentences are also shown here. This reinforces the idea that these trends are distinctive of truthful or deceptive text, however the increased difficulty of accurately tagging parts of speech in incoherent samples means that these results should not be taken with the same strength as that of the swing sentences. In particular, many of the standard deviations (with a small handful of exceptions such as base verbs ('VB') and prepositions ('IN')) are smaller in deceptive text than truthful text, again pointing to deceptive text being overall less varied. This lines up strongly with the results of the swing sentence analysis.

Generated Truthful Sequence

[CLS] can aliens aliens crimestellar aliens geek dinosaur armada nec aliens skulltsky ufo werewolf aliens cosmic aliens zombie aliens aliens titans predator predator police officers science lords battle armadabot predator chaos x spy warriors 3d police officers the aliens predator aliens zombie alien battleron aliens [SEP]

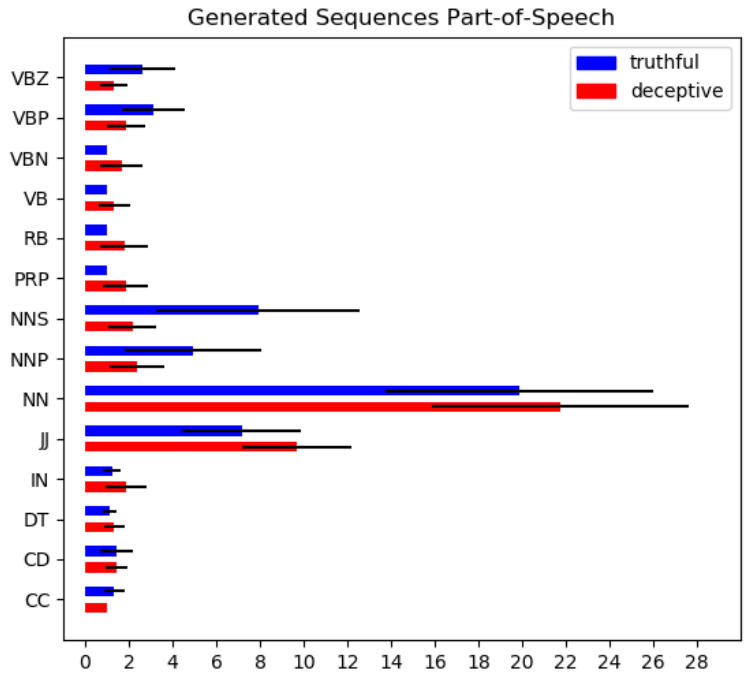

Figure 2.5: The part-of-speech analysis of the generated sequences. Red bars indicate deceptive sequences, blue bars indicate truthful sequences. Bar length indicates average number of occurrences per sentence with error bars representing standard deviation.

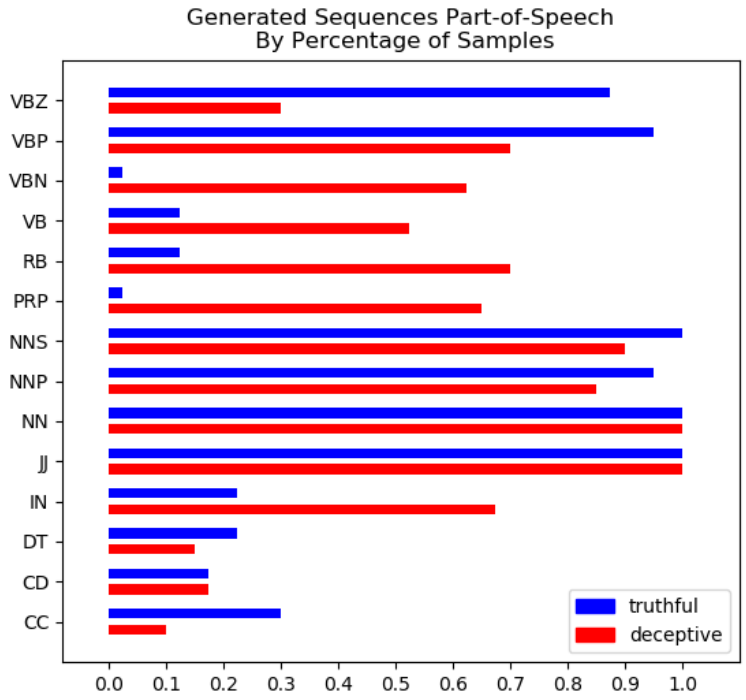

Figure 2.6: The parts of speech by the percentage of samples they appear in for generated sequences. Red bars indicate deceptive sequences, blue bars indicate truthful sequences.

Generated Deceptive Sequence

[CLS] aria me spaced reading vatro for tom want tom complete me league recording action tom , men tom ” league short tom complete tom march home quick with league drop russian short home tom quick reserve speech soon tom ” short tom ” cut short ! [SEP]’

The “at least once” appearances do not match the results of the swing sentences, but they are similar in that they both correspond to the mean appearances per sample. If a part of speech has a higher mean rate of appearance per sample, that same part of speech will also be prevalent in more samples. This suggests that it is not the specific part of speech that indicates truth or deception, but the variation in their use.

2.5 Discussion

In this project, I utilized BERT to understand what separates truthful text from deceptive text. BERT was able to beat state-of-the-art accuracy on the popular Ott Deceptive Opinion Spam Dataset. BERT's ability to reach this high accuracy indicates that features distinguishing truthful and deceptive exist and can be exploited. Our ablation study revealed that removing parts of speech such as singular nouns hurts BERT's ability to differentiate between truth and deception, indicating that certain word types are informative to the classification.

Our results with the swing sentences indicate that there is a high level of variation in truthful text, while the deceptive text was more formulaic in which parts of speech appeared. My Generative Adversarial Network produced similar results, reinforcing my conclusion that there are underlying patterns in deceptive text that do not appear in truthful text.

While some trends have been indicated this does not mean they are the sum total of BERT's self-created rules, and more can be done to expand BERT. I can modify the input, substituting phrases that are similar in meaning but different in language, which will allow me to see what can tip the classifier in one direction or the other.

More than that, I need to address the problem of data. The Ott dataset is an extremely useful gold-standard corpus, but it has flaws. It is small, for one thing, especially compared more recent datasets that can number in the hundreds of thousands. Beyond that however, is the issue of quality. The Deceptive Opinion Spam dataset is sourced from Amazon Mechanical Turkers following a simple prompt; there is no guarantee of effort behind the deception. If I want to delve deeper into my study of deceptive text, I need a new dataset, one that better reflects the attitude of people who are trying to accomplish a goal with their lie.

Chapter 3

The Motivated Deception Corpus

3.1 Introduction

I now have evidence to suggest that there are distinct differences between truthful and deceptive text. However, during the above project a question arose: how good was the Deceptive Opinion dataset as a corpus? It's a very well-organized and convenient dataset, after all there is a reason it is used as a gold standard in this area. But what if I could make something better? It was a common frustration to me that many existing corpora tended to be created along similar lines to each other, and thus had similar flaws. They tended to be either too small or too unreliable, either scraped from an online forum or filled out survey-style from someone uninvested in the process. It was in considering this issue with corpora that I decided to make my own advancement in this particular area.

One of the most basic requirements of any natural language study is the need for a quality corpus of data. Deception detection in text is no different. However, in this respect deceptive text has an additional complication that is not present in, for example, sentiment analysis. Sentiment in text is comparatively easy for humans to identify [Vogler and Pearl, 2019] and

generate, and good samples are relatively simple to gather. Quality samples of deceptive text, on the other hand, are much more difficult to assemble. Genuine deceptive text, by its nature, is made with the intent of fooling someone. This requires the deceiver to construct their deception in a manner that makes it look convincing. It is more difficult to source deceptive text samples in the wild due to the average human's poor ability to identify deceptive text [Levine, 2014, Ott et al., 2013]. It is possible to source deceptive samples in a traditional method of soliciting entries from subjects in exchange for compensation, such as with the Ott Deceptive Opinion Spam dataset [Ott et al., 2011], but this can lack the secondary factor of trying to make the deceptive text actively fool the reader. That is, while it is simple to obtain data that is *false*, there is little incentive on the part of the subjects to make the deception *convincing*, and therefore closer to a real-life sample. As Fournaciari et al observed, crowd-sourced online reviews are generally significantly different from wild examples [Fornaciari et al., 2020]. It is not necessary, after all, to put effort into making a convincing lie if all that is required of a subject is to produce an arbitrary sample.

To rectify this problem, I propose the Motivated Deception Corpus, with the goal of improving the quality of deceptive text through incentivizing higher quality deception. The incentive arises from the nature of the data collection, which takes the form of the game Two Truths and a Lie. This game revolves around the idea of presenting fellow players with a selection of stories, one of which is false. The other players must figure out which story is the lie while at the same time creating their own stories to fool other players. The structure of the game means that the player that is the best at making their lies believable and determining the lies of other players is the one most likely to win. By using a competitive structure, the subjects are motivated to make their deception convincing if they want to perform well in the game and thus be well rewarded. Using this technique, I have amassed a large amount of high-quality deceptive text to be used in natural language research. This corpus also reaches beyond the simple text and includes behavioral data as well. Every keystroke that the subjects made was recorded as they wrote the stories, including keystrokes that were

later deleted by the subject. The timestamps of the keystrokes in milliseconds are recorded alongside them as well.

We are familiar with deceptive text in the form of fake news and false reviews. These are often lumped together in one large category of deception. However, deceptions like fake news, while often expressed through the medium of text, are actually difficult to compare to occurrences like Deceptive Opinion Spam. The intent and purpose behind them, other than simply obscuring the truth, is usually wildly different. Fake news tells stories in a way that fake reviews do not. While some reviews can certainly contain a narrative, it is not essential to have one to qualify as a review. Fake news cannot operate this way beyond the basic reporting (or misreporting) of facts; a narrative is essential to qualify as news. My corpus is more similar to fake news or fake forum posts than false reviews, as both involve primarily a type of storytelling.

3.1.1 Machine Learning Efforts

Compared to classical lie detection, the amount of data for text-based deception is relatively small. Ott et al developed the Ott Deceptive Opinion Spam corpus, which consists of 800 true reviews from TripAdvisor and 800 deceptive reviews sourced from Amazon Mechanical Turk [Ott et al., 2011]. The Ott corpus is one of the most commonly used gold-standard corpora in deception detection tasks. However, it suffers from the fact that the Mechanical Turkers that created the deceptive samples were only asked and compensated for arbitrary samples; there was no additional incentive to be convincing. It is also, despite being of considerable size for a deceptive corpus, relatively small compared to corpora of other types. The Amazon Customer Reviews Dataset, by comparison, contains an enormous amount of data, on the order of a hundred million samples, but while some are undoubtedly false there is no easy way to identify them there is no deceptive label [Amazon, 2014]. There is also the DeRev dataset

made by Fornaciari et al, which identifies a number of helpful clues in detecting deceptive reviews, but still ultimately relies on human experts to identify said deception [Fornaciari and Poesio, 2014]. Wang et al created the LIAR dataset based on fake news as determined by Politifact [Wang, 2017]. This dataset is high-quality, as long as Politifact is reliable, but is difficult to scale, since it requires manual fact-checking of each individual story, and prone to subjective interpretations of political text. Other efforts in fake news, such as in Aphiwonsophon et al, use a variety of techniques including support vector machines, Naive Bayes algorithms, and neural networks to separate fake news from real [Aphiwongsophon and Chongstitvatana, 2018]. Feng et al assembled a dataset of 800 true and false reviews identified by Yelp's filter system, however Yelp's criteria for identification are not publicly disclosed [Feng et al., 2012].

There are a few corpora that while not true deceptive text corpora, are similar enough in premise to make them worthwhile to explore when examining deceptive text. Filatova et al created a corpus of Amazon reviews that were labeled as ironic (or sarcastic) or normal [Filatova, 2012]. These reviews were gathered and labeled by Amazon Mechanical Turkers and number 1254 samples in total. While these samples share qualities with deceptive text in that both are not presenting the unvarnished truth, it is difficult to use this corpus as a deceptive corpus. The labeling is crowdsourced from non-experts and can be highly subjective. Furthermore, irony cannot safely be put in the same category as true deception as while neither is technically telling the truth, irony and deception have inherently different goals of intention. Irony is meant to be noticed, difficult though that is in a text setting, while deception is not. The Self-Annotated Reddit Corpus (SARC) of sarcasm by Khodak et al is made of 1.3 million Reddit comments, which are labeled by the commenter [Khodak et al., 2017]. The size of this corpus is impressive, although it relies on the Reddit user submitting the comment to self-report the sarcasm. In this corpus, a comment ending in a '/s' is flagged as sarcastic and one without that ending is genuine. This allows for mass scraping of Reddit comments, but this nomenclature is not universally followed. Like Filatova et al's corpus, it

also cannot be directly used as a deceptive corpus due to the inherent differences between true deception and sarcasm, although some techniques can be applied to both types of text.

Previous attempts to perform deception detection often rely on techniques such as support vector machines and linguistic characteristics. Vogler et al used a support vector machine operating on linguistically defined features to classify the Ott corpus [Vogler and Pearl, 2019]. They were able to achieve an accuracy of 87% using this method. Xu et al train a maximum entropy model on the Ott corpus and were able to achieve 91.6% accuracy [Xu and Zhao, 2012]. Li et al tried to find a general rule for identifying deceptive opinion spam using features like part-of-speech on several datasets including the Ott corpus, achieving 81.8% accuracy on Ott [Li et al., 2014]. Ren et al expand on this work by using a recurrent neural network on the same data, improving the accuracy to 85.7% [Ren and Ji, 2017].

On the neural network side, several interesting tools have arisen to process textual input. Hu et al used a variety of models to identify concealed information in text and verbal speech, best among them a deep learning model based off bidirectional LSTMs [Hu, 2019]. Concealed information, in this context, refers to when a person has knowledge about a subject and is withholding it, as compared to Hu's definition of deception where someone fakes knowledge they do not have. Hu created a corpus of wine tasters evaluating wines and encoding in various ways such as n-grams, LIWC, and GloVe embeddings [Pennington et al., 2014] based on the recordings. The LSTM model using these features achieved an f-score in identifying the presence of concealed information of 71.51, defeating the human performance of 56.28.

One of the standout neural network models in working with text-based tasks is the Bidirectional Encoder Representations from Transformers (BERT). The groundwork for this model was laid by Vaswani et al, who developed a new kind of network based on self-attention that showed dramatic improvements in the area of machine translation [Vaswani et al., 2017]. Devlin et al built on this work to create the structure known as BERT, using many instances of self-attention networks to learn contextual representations of text. BERT has proven highly

competitive in multiple areas including sentiment classification and next sentence prediction [Devlin et al., 2018]. Barsever et al were able to utilize BERT in order to perform deception detection on the Ott dataset, proving its viability and power in that arena and setting the state of the art [Barsever et al., 2020]. Their model was able to achieve an accuracy of 93.6% on the Ott Deceptive Opinion Spam corpus. They also used BERT as a generative model to produce machine-created samples of both truth and deception, and identified some linguistics trends such as deceptive samples being less varied in the parts of speech that they used.

3.1.2 Our Corpus

With this Motivated Deception corpus, I introduce a set of deceptive text that is large, realistic, and challenging. This will provide new avenues and benchmarks to researchers working in the field of deceptive text. It includes both raw text and behavioral data in the form of the keystroke timestamps, combining textual data and behavioral data.

In addition to the corpus itself, I provide several machine learning benchmarks on the accuracy and hit rate of the text in several configurations. These should form useful guidelines for researchers aiming to evaluate the efficacy of their own methods on this corpus. None of these experiments were formally preregistered. The data collected is available here.

3.2 Methods

Data was sourced from 177 University of California, Irvine undergraduate students between 18 and 29 years of age using the Experimental Social Science Laboratory Sona system at UCI. 17 subjects were determined to be operating in bad faith (submitting mutually exclusive 'true' stories and other suspicious behavior) and were removed as bad actors, for a total of

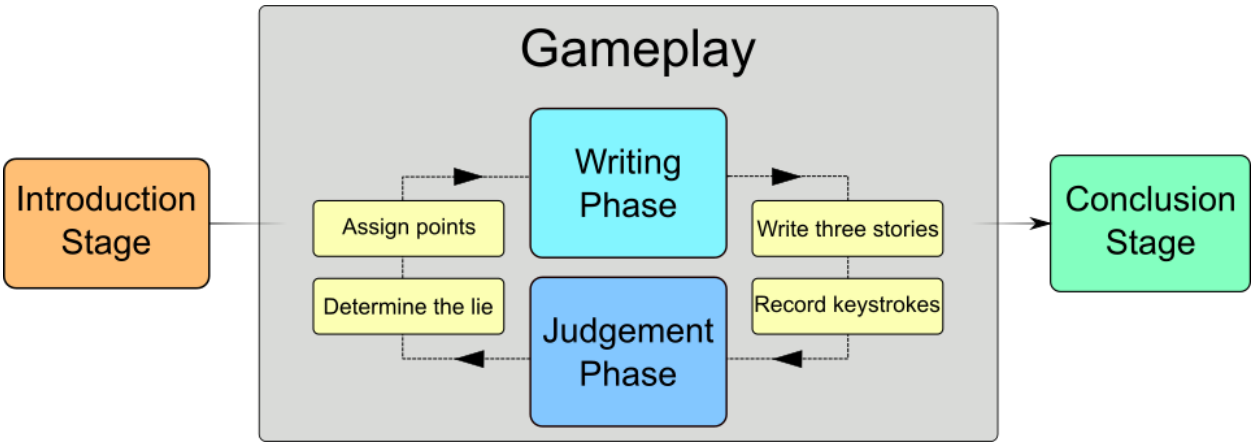

Figure 3.1: The general flow of the Two Truths and a Lie game. First is the introduction phase where the mechanics are explained, followed by the main gameplay loop where stories are written, recorded, and judged. Finally the scores are tallied in the conclusion and compensation is given to the player.

160 subjects. The game was developed in oTree Studio [Chen et al., 2016] and deployed on a Heroku server in order to make it accessible from any computer with a link to the study. All subjects were paid a $7.00 show-up fee as well as additional rewards based on how they performed in the game. Sessions were online and consisted of between 20 and 50 subjects participating synchronously at their computers with a mouse and keyboard. All instructions were text-based. The time remaining for each page was clearly visible on all pages. All data collection was done remotely with a researcher monitoring each session but otherwise not interacting with the subjects. If a subject needed to leave early, the researcher would take over their position in the game and the subject would be compensated for the work they did up to that point. Any data entered by the researcher, as well as entries that were blank or obviously violated the spirit of the game (such as one-word entries and "true" stories that were mutually exclusive) were discarded.

3.2.1 Two Truths and a Lie

The technique used to gather data for this corpus is based on the game Two Truths and a Lie. The game consisted of an introduction stage, ten rounds of gameplay, and a conclusion stage. At the end of the game, subjects received points based on how well they performed during the game and are compensated proportionally. A general outline of the game can be seen in Figure 3.1.

The Introduction Stage

In this phase, subjects are given written instructions on how to play the game, as well as an explanation of the reward system. Subjects could indicate their readiness to move on to the next phase by pressing a button at the bottom of the page, or they would be automatically advanced after 5 minutes. You can view the screen the subjects saw in Figure 3.2.

Gameplay

Subjects participated in 10 rounds of gameplay corresponding to 10 provided topics ("Homework", "Vacation", "Dinner", "Exams", "Housing", "Dating", "Shopping", "Family", "Friends", and "Fitness"). Each round of gameplay consisted of two phases, a writing phase and a judgement phase.

In the writing phase, subjects were presented with the round's topic and a text box within which to write a story between 30 and 200 words long. When finished writing the story, they would press a button and move on to the another page with a new text box. In this manner, subjects wrote true stories on the first two pages and a lie on the third. Each page would automatically advance after two minutes, and any data entered would automatically be submitted. During this phase, the subject's keystrokes would be recorded, as well as the

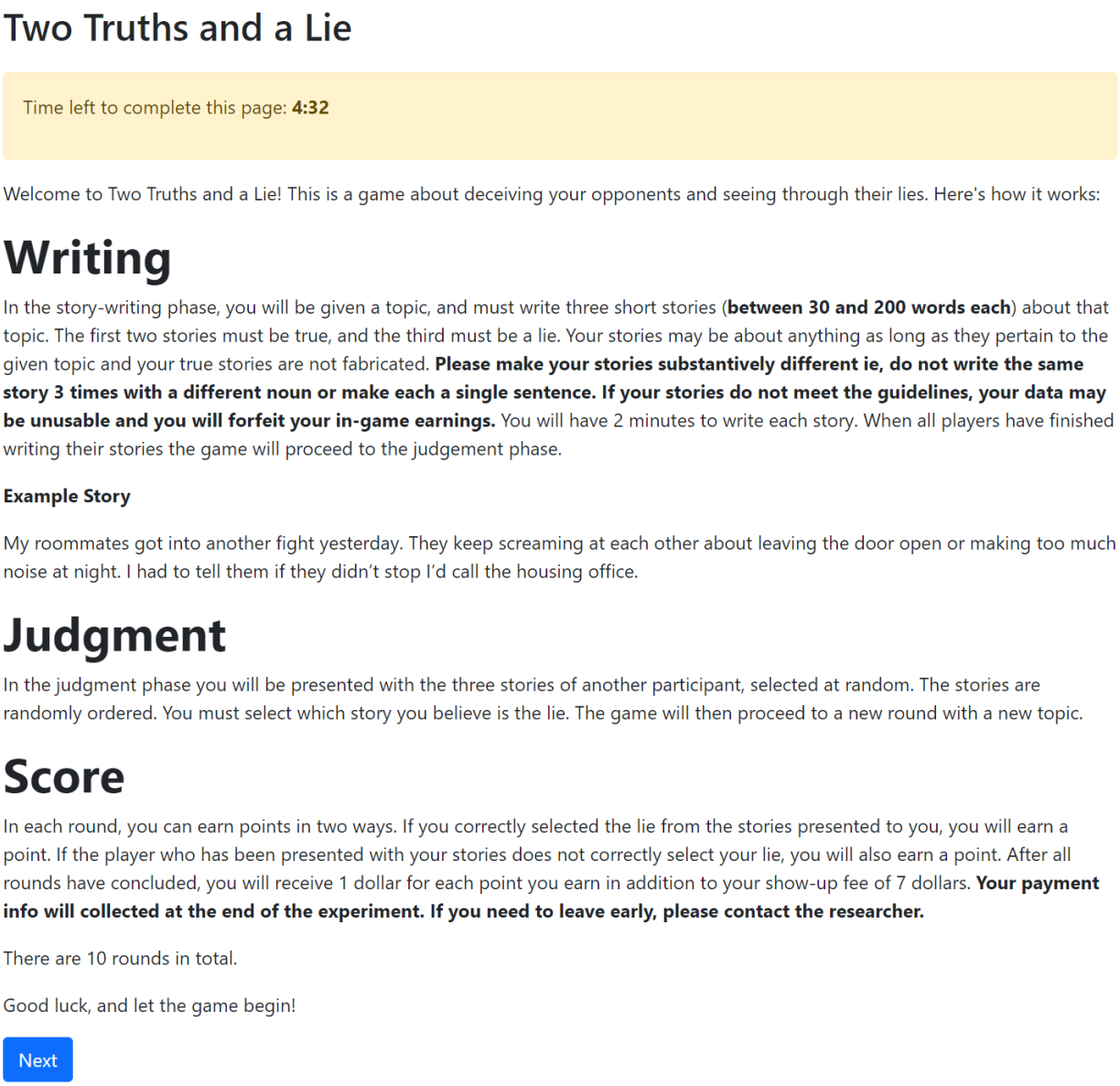

Figure 3.2: The introduction screen to Two Truths and a Lie.

timestamp of the keystroke in milliseconds after the page was loaded. Subjects were not informed about this aspect of the data collection. After writing the false story, gameplay would proceed to the judgement phase. An example story being written is shown in Figure 3.3.

In the judgement phase, each subject's trio of stories would have its order randomized and sent off to another random subject, so that each subject is viewing one other subject's story

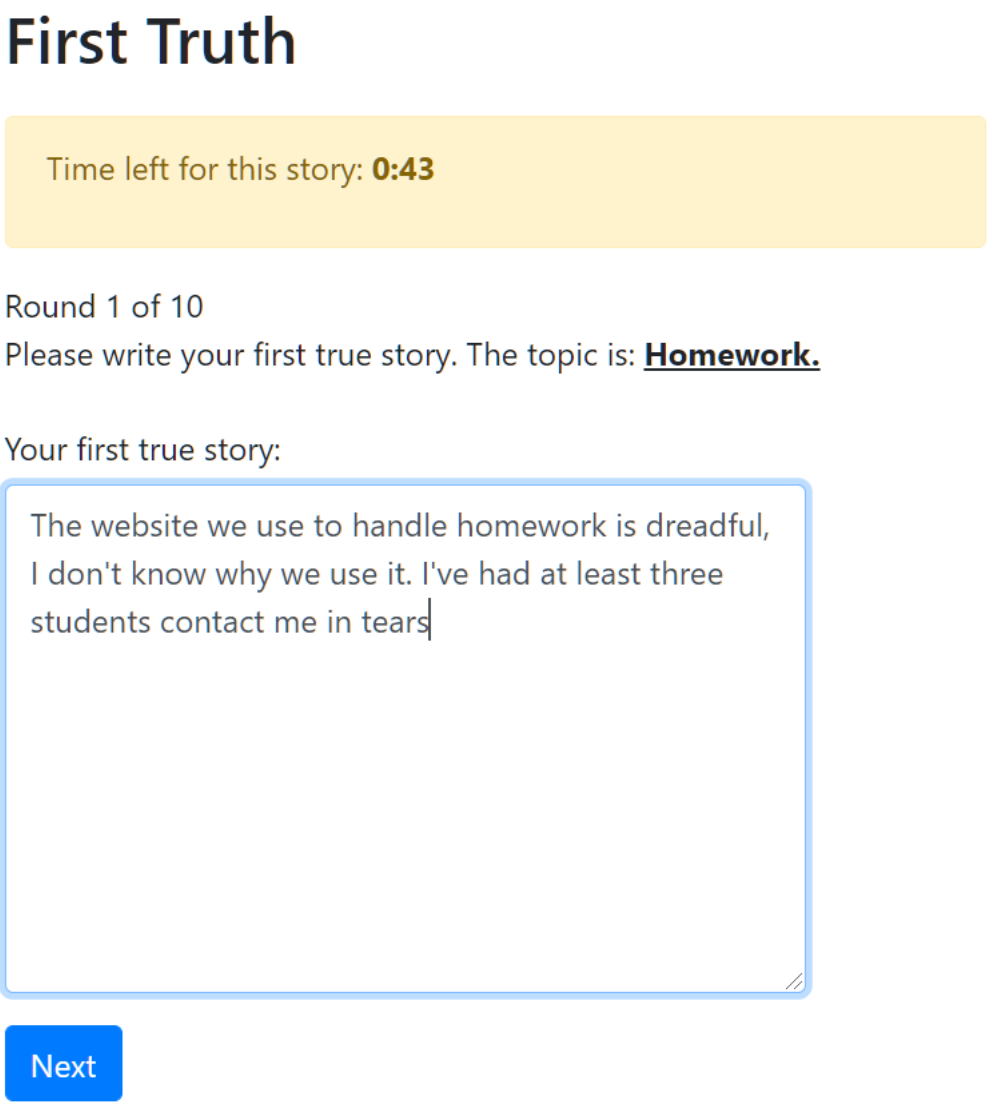

Figure 3.3: A participant writing a true story.

trio. All stories in the trio were presented at the same time. The order of the trio was also shuffled so that the receiving player cannot simply use the position in the list to make the determination. The subject was given two minutes to read the stories and select via button press which one they believed to be the lie. This equates to a 3-alternative forced choice task where the signal being determined is the lie. If no selection was made in time, the page advanced automatically and no choice was recorded. If a subject correctly guessed which of the trio was the lie, they received a point. If a subject guessed incorrectly or did not make a choice, the subject who wrote the trio received a point instead. After all judgements are

Table 3.1: Example triplet submitted by the subjects. The topic for the round was 'Family.' The player receiving this triplet correctly identified the lie.

Label	Story
Lie	I recently went to my relative's wedding in Philadelphia. It was a fun experience overall. While I was there I was able to go see the Liberty Bell which was cool.
Truth	I have a large family. I have 3 younger siblings and I'm the eldest sibling. Because of this I have added responsibilities and watch my siblings when my parents aren't home.
Truth	All my grandparents but one died before I was born. I don't know my living grandparent too well because he lives in another country and doesn't speak English.

Table 3.2: Example triplet submitted by the subjects. The topic for the round was 'Family.' The player receiving this triplet failed to correctly identify the lie.

Label	Story
Lie	I have two cousins from my father's side. I honestly don't understand why they hate me. I haven't spoken to them since I was five.
Truth	I have twelve cousins from my mother's side of the family. None of which I am close to.
Truth	I am currently quarantined with my mother and sister which I am really thankful for.

recorded gameplay proceeds either to the next round or the conclusion phase if all rounds are completed. An example of the judgement screen is shown in Figure 3.4.

Conclusion and Payment

In this stage, each subject's points were tallied up and the subject was told how many points they earned. Subjects could earn between 0 and 2 points per round, totalling to between 0 and 20 points over ten rounds. Each point equated to an additional dollar of reward money. Each subject received a personal code to be inputted in a compensation form as well as a link to post-game survey where they could give feedback on the experiment. Subjects were primarily paid through Venmo, Paypal, and Zelle. An example conclusion screen is shown in Figure 3.5.

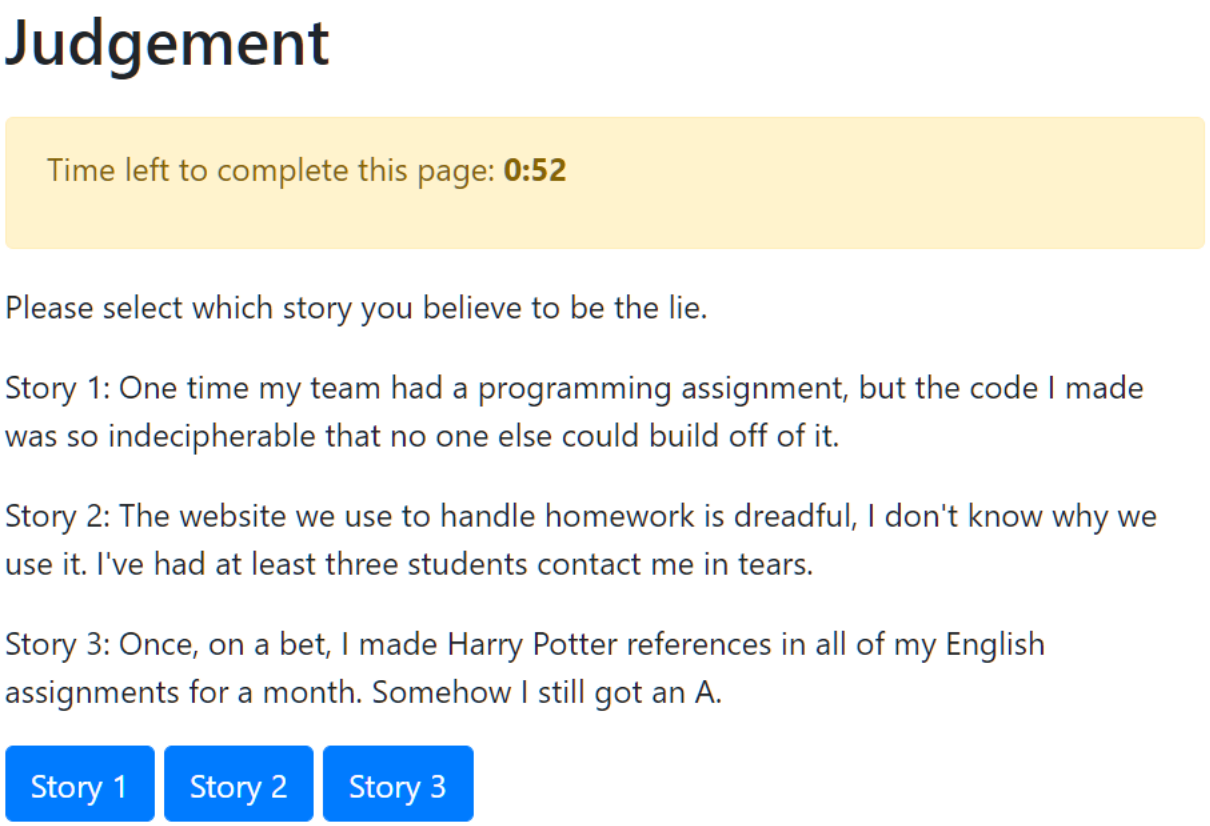

Figure 3.4: A participant judging another player's entries in Two Truths and a Lie. The third story is the lie.

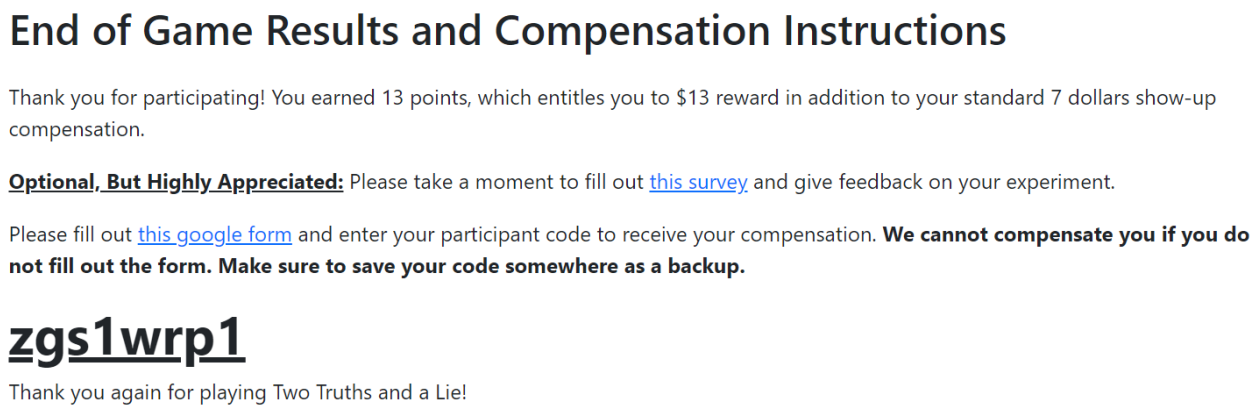

Figure 3.5: An example conclusion screen in Two Truths and a Lie.

3.3 Results

3.3.1 Corpus

After playing Two Truths and a Lie with 177 subjects, I have assembled a corpus of deceptive text that is both large and realistic. The linguistic makeup of this corpus is outlined in Tables 3.3 and 3.4. The average parts of speech per story (shown in Table 3.4) were defined and tagged with the Natural Language ToolKit [Loper and Bird, 2002] after each story was tokenized into words. In general, lies tend to be slightly longer with more variance in the length, although the variance in both categories makes it difficult to classify based on that alone. Note the example in Table 3.2, in this case the lie is significantly longer than either true story, and it was not identified as such. By contrast, Table 3.1 shows a lie that is of similar or slightly less length than the true stories, but it was correctly identified. This indicates that subjects do not automatically identify longer stories as deceptive. Both categories also tend to be similar in terms of the part-of-speech makeup, as shown in Table 3.4. The biggest difference is in the amount of interjections, where true stories have almost twice as many occurrences.

After discarding unusable data, I acquired 1572 deceptive samples and 3144 truthful samples along with their corresponding keystroke data. The truthful stories had a mean length of 30.17 words with a standard deviation of 17.95 words. The deceptive stories had a mean length of 31.25 words with a standard deviation of 19.03 words.

Table 3.3: Descriptive statistics of the stories generated.

Statistic	Value
Average Length Truth (words)	30.08
Standard Deviation Length Truth (words)	17.74
Average Length Lie (words)	31.29
Standard Deviation Length Lie (words)	19.05

Table 3.4: The prevalence of given parts of speech in true and false stories. All forms of adjectives, adverbs, verbs, and nouns are grouped together.

	Average percentage of story	
Part of Speech	Truth	Lie
Coordinating Conjunction	3.275	3.279
Cardinal Digit	1.489	1.259
Determiner	6.827	7.079
Preposition	10.449	10.483
Adjective	6.120	6.216
Modal	0.728	0.741
Noun	18.833	18.849
Predeterminer	0.0877	0.0894
Possessive Ending	0.154	0.163
Possessive Pronoun	3.379	3.669
Adverb	6.574	6.449
Particle	0.785	0.767
Infinite Marker (to)	2.979	3.177
Interjection	0.0106	0.00610
Verb	18.321	18.415
wh-Determiner	0.209	0.215
wh-Pronoun	0.187	0.183
wh-Adverb	0.735	00.671

3.3.2 Human Performance

Two Truths and a Lie was not only about recording samples of text, but observing the ability of the human players to judge each others' samples. In this regard, they were generally ineffective. The average accuracy of the players when judging samples was 35.7%, close to random chance (33%). Using a 95% confidence interval of proportions, I calculated the true mean accuracy of the human population to be between 26.8% and 42.8%. This leaves me unable to conclude that humans are, on average, better than random chance at identifying deception.

I also calculated the sensitivity (d') of each subject. The sensitivity of the subject is a measure of how well they can identify deception when it is present. The sensitivities were derived from the table outlined in Frijters et al using the metric for a 3-alternative forced

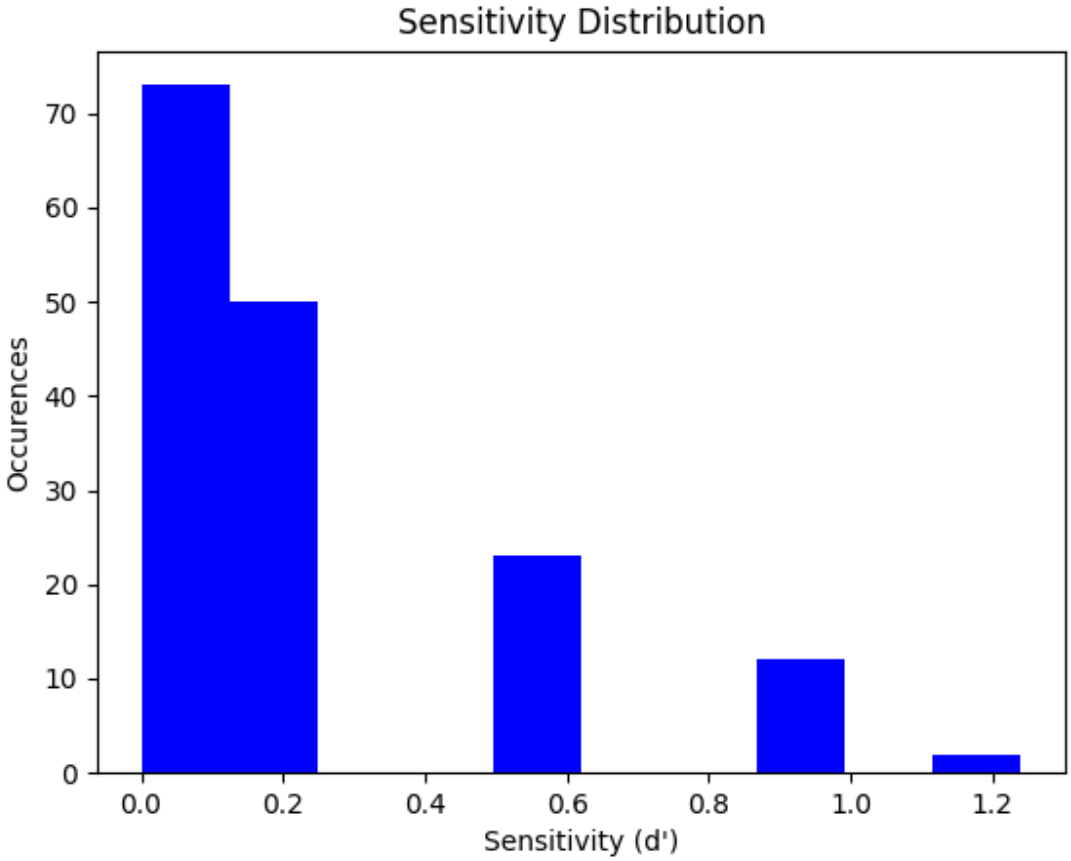

Figure 3.6: The histogram showing the distribution of sensitivities across the subject pool. Players that scored at chance or below were given a sensitivity score of 0.

choice task [Frijters et al., 1980]. In total, 45.6% of players were assigned a sensitivity of 0, meaning that they performed at the level of random chance or worse. Only 23.1% managed a sensitivity score better than 0.5. This reinforces the conclusion that the ability of humans to detect truth from lies is weak. A histogram showing the full distribution of sensitivities is shown in Figure 3.6.

3.3.3 Machine Learning Benchmarks

I applied neural networks on the corpus to establish some machine learning benchmarks and compare them to benchmarks on other corpora. A summary of the results can be seen in

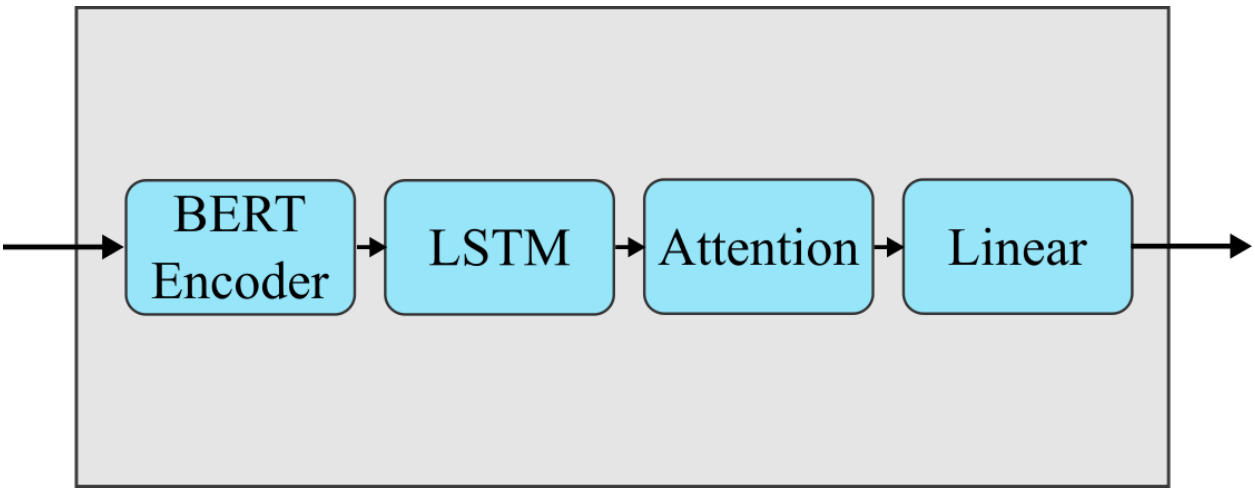

Figure 3.7: The structure of the discriminator used to classify the corpus data. The standard BERT model is followed by a bidirectional LSTM, a self-attention layer, and a fully connected linear layer.

Table 3.5. The base classifier I used was the one utilized in Barsever et al [Barsever et al., 2020]. The structure of this classifier is shown in Figure 3.7. The first layer of the classifier is the pre-trained BERT model from the huggingface transformer library [Wolf et al., 2020]. The pooled output is extracted from the BERT model and fed into a bidirectional recurrent LSTM layer. The output from this recurrent layer is then fed into a self-attention layer based on the machine translator networks from Vaswani et al [Vaswani et al., 2017]. Finally the attention layer output is passed through a fully connected linear layer that classifies the input into two possible classes: truthful or deceptive.

I chose this network because of its power and ease of use. BERT, or Bidirectional Encoder Representations from Transformers, performs well in a wide variety of contextually sensitive language tasks due to being able to detect when the meaning of a sequence has changed depending on context. This allows it to detect subtle differences in phrasing [Devlin et al., 2018]. It also requires very little preprocessing of the data, allowing samples to be fed directly into the model without adding additional steps or complexity. The BERT-based network achieved a 93.6% accuracy on the Ott Deceptive Opinion Spam dataset, making it the state of the art in the field of deception detection. By comparing its results on the Ott dataset

Table 3.5: Performance statistics of the BERT classifier.

BERT Model	Accuracy	Hit Rate	False Positive	Sensitivity (d')
3-AFC	41.2%	N/A	N/A	0.260
Binary Classification	57.8%	40.5%	33.5%	0.188
NSP* (Unshuffled)	79.9%	77.4%	17.3%	1.694
NSP (Shuffled)	56.0%	67.2%	55.03%	0.319

*Next Sentence Prediction

with the results achieved on the corpus I made, I can gain a rough idea of how much more challenging my corpus is to classify.

BERT Configuration 1: 3-Alternative Forced Choice

I tested several configurations of the BERT classifier on the corpus. The configuration I refer to as BERT for 3-Alternative Forced Choice, or 3-AFC BERT, is the most directly analogous to the human task. In this configuration, each story in a particular triplet is presented to the classifier in turn, and the strength of the 'lie' class neuron is recorded for each one. The story that generated the strongest 'lie' reaction is considered the model's 'choice,' allowing the network to 'pick' from the three stories in a simulacrum of the humans' 3-AFC task from the game. This configuration was able to reach an accuracy of 41.2%, outperforming the human subjects, but not significantly. Due to the nature of a 3-AFC task, I cannot calculate a rate for hits or false positives, but using the table from Frijeters et al I find the sensitivity to be 0.26.

BERT Configuration 2: Binary Classification

The next configuration is a more traditional machine classifier, with simple binary classification. In this mode, BERT is presented with each story individually and attempts to make a true/false classification without any additional context. This configuration was able to achieve an accuracy of 57.8%, however since the number of samples in each class is unbal-

anced (there are two truths for every lie) it can be difficult to evaluate the network using only accuracy. When looking at the hit rate and false positive rate, I find that the network managed 40.5% and 33.5% respectively, with a sensitivity score of 0.188. This indicates that the binary configuration is less sensitive than the 3-AFC setup.

BERT Configuration 3: Next Sentence Prediction

I also utilized another form of BERT, BERT for Next Sentence Prediction, to perform a slightly modified classification task. This model differs from the base BERT model in both the style of its inputs and the meaning of its output. This model takes two sequences as inputs. The first of these is the *context*, or the 'first sentence'. The second sequence is referred to as the *query*, or the 'next sentence.' The task of the model is to predict whether the query was the next sentence following the context in the original document. At its core, this is still a classification problem, with one class being 'yes this is the next sentence' and the other class being 'this sentence is unrelated.' I used this basic structure to simulate, to a limited extent, the network playing the game. I first divided the corpus into each triplet of stories submitted by the players. The stories are then assembled into pairs. Each pair consists of one of the true stories as the context and either the remaining truth or the lie as the query. With two truths available to act as contexts and each having two possible queries, this creates four paired samples for each triplet. This has the side effect of balancing the size of each class, as there now an even number of cases where the lie is the query. The network then runs its next-sentence-prediction classification, training on whether or not the query is a lie given its context.

I ran this configuration on two versions of the dataset, which I refer to as shuffled and unshuffled. In the unshuffled version, the query and context are restricted to being from the same triplet. In the shuffled condition, the query and context are selected from separate, randomly selected triplets. I fine-tuned this model on both the shuffled and unshuffled

versions of this corpus of paired samples for 100 epochs.

Under the unshuffled condition I achieved an accuracy of 79.9%, a hit rate of 77.4%, and a false positive rate of 17.4%. This is the highest performing configuration so far, indicating that BERT can learn contextual clues for a given triplet. However, when the stories are shuffled, all of these measure worsen, particularly the accuracy and false positive rate, which change to 56% and 55% respectively. The dramatic drop in performance indicates that BERT cannot generalize cues across different subjects.

3.4 Discussion

Detecting deceptive text is a difficult task, made even more onerous by the lack of comparability between crowdsourced data and real-world data. Between fake news, misleading forum posts, and sock puppet online reviews, the opportunities for deception only continue to grow. With the ability of online deceptive text to propagate and influence real-world decisions of readers, it is important to have a corpus of data that accurately reflects the level of effort of what a reader is likely to find online. I created a Motivated Deception corpus of text samples that is designed to be closer to real life deception than other corpora. I did this by incentivizing players of Two Truths and a Lie to both deceive and perceive deception on their fellow players by paying them based on their performance. By turning the collection process into a game, it allows the data collected to be more indicative of real-world samples of deceptive text like fake news and false social media posts. This has created a corpus that is large, realistic, and challenging for both machine classifiers and humans.

The storytelling involved in the game necessitates a caveat when examining this corpus. When creating this corpus, which stories were true and which were lies was self-reported by the participants. While I eliminated several obvious bad actors, it is impossible to verify

every subject's story as genuinely truthful or deceptive. The subjects were aware that the stories could be vetted and their credit could be revoked, but it is still possible that some could have attempted to cheat the system. This is a problem endemic to data collection of this kind, however even with this caveat I believe this corpus is a valuable resource for those looking for motivated samples of deception. The oversight of a researcher during the live data collection and interactions with the subjects indicate that the vast majority operated in good faith and gave genuine entries while playing the game. In general, online subjects tend to be honest when self-reporting [Paolacci and Chandler, 2014], and with few exceptions there has been no reason to assume any different from the players of this game.

The first and most relevant benchmark to assess is how accurately do humans perform the task of identifying deception when they see it. I found that overall the humans subjects performed at around the level of random chance, with almost half having a sensitivity score of zero. This is consistent with the goal of incentivizing the players to produce convincing deception, and thus making the task more challenging, as well as the natural poor ability of humans to recognize deceptive text.

I used the state-of-the-art BERT classifier to generate machine learning benchmarks on this corpus in order to help quantify the level of difficulty compared to previous corpora. I designed several configurations of BERT to provide benchmarks in a variety of contexts. One such configuration was the 3-AFC configuration, presenting each story belonging to a given trio to BERT and recording the output of the 'lie' classification neuron, simulating the 3-Alternative Forced Choice the human subjects faced. This configuration resulted in an accuracy of 41.2%.

In a binary classification task, BERT achieved an accuracy of 57.8%, a hit rate of 40.5%, and a false positive rate of 33.5% on this corpus. This is an improvement over the human performance. It does, however, indicate that the network has a preference for truth, given that the miss rate is significantly higher than the false positive rate. This is perhaps to be

expected, given that the number of truthful samples is double that of the deceptive samples. When the data was sampled to have equal quantities in both classes I achieved an accuracy of 53.9%, compared to the 93.6% accuracy it achieved on the Deceptive Opinion Spam dataset, marking it as significantly more difficult to classify. When classifying on the keystrokes or their timestamps, the accuracy is no better than chance.

Another configuration used the BertForNextSentencePrediction model from huggingface's transformer library. When using one truthful statement as the context and one other statement from the same trio as the query, the model was able to predict whether the query was truthful or deceptive 76% of the time. This accuracy only exists when predicting on statements from the same triplet however; when the context and query are shuffled so that they are from different triplets the accuracy drops closer to chance (56%).

All the configurations of BERT struggled with this corpus, much more so than when it was used to evaluate the Ott Deceptive Opinion Spam dataset. This shows that my corpus is significantly more challenging than previous corpora. It is my intent that the benchmarks I establish here not be taken as endorsements of my model, but starting lines so that whoever uses this corpus will have a baseline with which to compare their results. I aim for this corpus to drive the creation of more sensitive, nuanced models that can capture the intricacies of realistic deception.

Exciting opportunities unfold with the creation of this corpus. I have examined the behavior of machine learners on deceptive text, and now I have built a corpus of my own. It is now time to turn to the thorny area of human perception of deceptive text. In other words, after examining the classification of deceptive text and the generation of deceptive text, it is now time to examine the judgement of deceptive text. I will turn to the old standby, online crowdsourcing, and use the corpus created here to see just what underlies the decision of a real person to label a sample as truthful or deceptive. I will shift focus away from simple accuracy and examine bias in the decision making process. The question at hand now is not

how good humans are, but what makes them think a piece of text is one class or the other. It is with this in mind that I develop the third and final project in this research.

Chapter 4

Human Judgement of Deception

In the first project I looked at how a machine views deception and established some basic ground truths. My second project delivered us a brand-new corpus of realistic deceptive text. It is now time to examine the ever-mercurial realm of human judgement.

Ask someone why they believe their best friend or poker partner was lying, and you can receive a multitude of reasonable answers. Perhaps they were fidgeting, or sweating, or refusing to make eye contact. Even something like keeping your head still can be considered a tell by poker players [Mandjes, 2019]. These signs of stress are commonly taken as indicators of deception. The practice has a long pedigree; interrogators in ancient China would fill the mouths of subjects with dry rice, acting on the premise that lying would cause dry mouth and therefore not wet the rice [Ford, 2006]. Other signs of stress that are believed to indicate lies are responses such as increased heart rate or respiration. All of these and more are intuitive ways that humans perceive lies person-to-person, and more importantly are known metrics of deception. Their presence or absence is so well-known that many polygraph tests are based on these cues [Council et al., 2003]. But what about text-based lies, where there are no such cues?

Every time someone reads an online review or scrolls through a news article, they are making a decision about the content of that article, even if only subconsciously. These decisions often have significant ramifications on the lives of the reader. People make economic decisions based on the online reviews they read of the products they are interested in [Murphy, 2018]. The narratives in a fake news article can affect their political beliefs, which can have knock-on effects in their judgements of other things. Exposure to misinformation about vaccines, for example, can reduce the consumer's probability of getting inoculated [Carrieri et al., 2019]. Even something as simple as reading a message board or a chat with a friend can influence the reader into changing their behavior. The thing is, we already know that humans are generally terrible at identifying deceptive text ([Levine, 2014, Ott et al., 2013]). The question I am interested in is: what do humans consider truthful or deceptive when it comes to text? This is not a question of how accurate humans are at judging deceptive text. Rather, it is a question of what makes someone believe that a given sample is truthful or deceptive.

To answer this question, I presented online users of Amazon Mechanical Turk with samples of truthful and deceptive text and asked them to judge which ones were deceptive. These samples were randomly selected from the Motivated Deception Corpus [Barsever et al., est 2022] and consist of triplets of stories from the party game Two Truths and a Lie. Each triplet contains one false story and two true stories. I asked the Mechanical Turkers to select which story among the three was the lie, indicating how confident they were in their decision. They also had the option of noting their reasoning for the picking a particular story in a feedback text box.

I then analyzed the responses to determine any biases that might be influencing the choices of the Turkers. I do this by observing the part-of-speech makeup of the stories, the presence or absence of certain words or types of words, and applying neural network classifiers to the stories. The neural networks I utilize are variations of BERT (Bidirectional Encoder Representations from Transformers) [Devlin et al., 2018]. BERT is constructed from the

transformer networks pioneered by Vaswani et al and performs very well in a variety of language tasks [Vaswani et al., 2017]. My main classifier is the discriminator constructed in Barsever et al [Barsever et al., 2020], but I also make use of other configurations from the huggingface transformers library [Wolf et al., 2020].

I have found that there is a definite bias that can't be explained by random chance, but identifying that bias in a quantitative sense is not a trivial task. The direct feedback from the Turkers identify qualities such as the vagueness or plausibility of the story, however this is difficult to translate for use in any kind of algorithm. The Turkers' judgements can't be separated by the sentiment of the story, the length of the story, the presence of numerals, or the concentration of proper nouns. I did find some evidence that the Turkers consider words that indicate extremeness and negativity to be indicative of lying.

What is more definite is presence of certain combinations of words. The bigram 'I was' is more prevalent is the stories perceived as deceptive (the *dominant* stories) than the ones perceived as truth (the *avoided* stories). In fact, in dominant the deceptive stories have more collocations involving past tense verbs than the avoided stories. This leads us to an interesting find: when it comes to difference between perceived truth and perceived deception, the presence of a *narrative* in the text might be an important factor.

4.1 Methods

I used Amazon Mechanical Turk to gather data for this research. I accomplished this by building a survey-style experiment in oTree Studio hosted on a Heroku server. Turkers had to meet three requirements before they could accept the experiment. 1) They had to be located in the United States. 2) They had to have completed at least 100 HITs (Mechanical Turk experiments). 3) They had to have a HIT approval rating greater than 95%. Subjects

were paid a $1.00 flat fee as well as a bonus based on performance. Subjects were not allowed to repeat the experiment. The experiment had three stages: Introduction, Judgement, and Conclusion.

The content for the game was extracted from the Motivated Deception Corpus of story triplets from the game Two Truths and a Lie. Each triplet contained one lie and two true stories. 300 triplets were randomly selected to serve as the triplet pool for this study. From this pool, each subject was presented with a random selection of 37 triplets to judge along with 3 custom-made triplets that served as attention checks.

4.1.1 Stage 1: Introduction

In this stage the subject is informed of the nature of the experiment and what would be expected of them. When the subject was ready to proceed, they would press a button and move on to the next stage.

4.1.2 Stage 2: Judgement

In this stage the subject is presented with a randomly selected triplet of stories from the Motivated Deception Corpus, emulating the game Two Truths and a Lie. The subject is tasked with selecting the one lie among the triplet. They would also indicate their confidence level for each response, selecting from the options of Just Guessing, Somewhat Confident, Reasonably Confident, and Certain. There was also an optional free-response box for subjects to explain why they made the choice they did. For every lie successfully spotted, the subject would earn a bonus of $0.50. This stage looped for 40 rounds. Subjects were informed of whether or not they selected the correct story, but not shown the correct answer if they got it wrong. There was no time limit to submit a response. An example of the page seen by

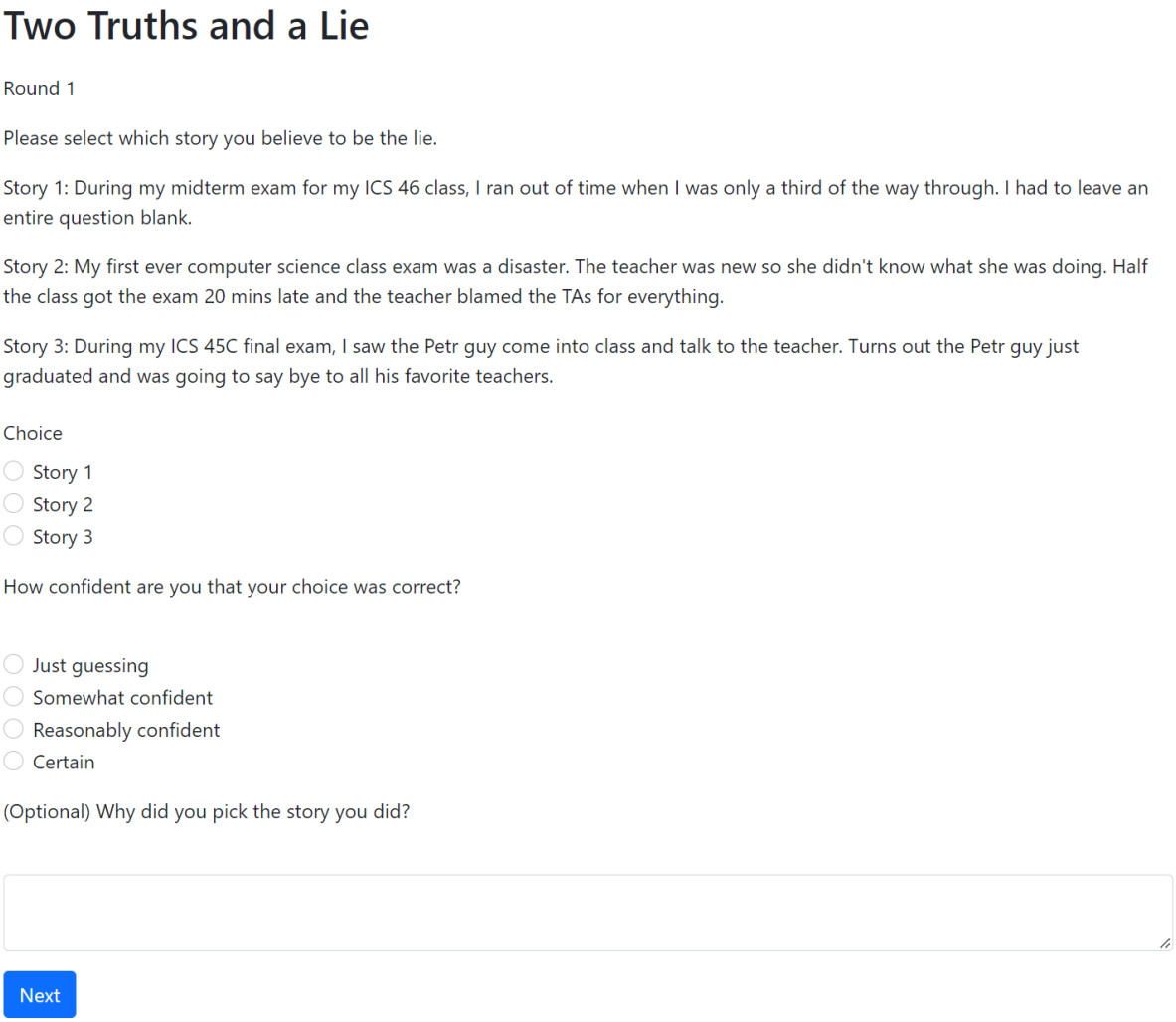

Figure 4.1: An example triplet seen by a Mechanical Turker. The correct answer is Story 3.

the Turkers is shown in Figure 4.1.

Attention Checks

Mixed in with the normal triplets were three sets of attention checks. These consisted of stories constructed in such a way that it would be a trivial task to spot the lie among the truths. The attention checks were varied in length, detail, and list position so that there was no pattern to the correct answer. If a subject selected the wrong option in these special triplets, they would be marked as having failed the attention check. The attention check

Attention Checks
"I graduated high school."
"For my middle school science fair, I built a time machine and went back to kill the dinosaurs."
"I wear shirts."
"My house has doors."
"At some point in the past, I ate breakfast. And then later, I ate more stuff."
"I was the first man on the moon. Not Neal Armstrong. Me."
"I typed this on a keyboard."
"Normally I sleep for less than 30 hours at a time."
"I'm not actually a human. I'm a sentient lion that learned to type, and one day I will have revenge on the scientists that created me."

Table 4.1: All attention check triplets used in this study.

triplets I used are in Table 4.1. They did not vary between sessions or participants, but since Turkers were not allowed to take this HIT twice it was not an issue.

4.1.3 Stage 3: Conclusion

In this stage subjects were informed of how many points they earned in total and given a code to submit through Mechanical Turk. This code indicated to the researcher that the subject had completed the study and was ready to be paid.

4.1.4 Bayesian Cultural Consensus

Our aim is not to establish whether humans are accurate, but to identify patterns of bias in the result. To this end, I employ the Bayesian Cultural Consensus (BCC) model, with some modifications to fit the needs of the project. The original Cultural Consensus Theory model can be summed up as the idea that, given a set of student answers to a test, it is possible to reconstruct a missing answer key to that test [Weller, 2007]. I am not reconstructing

the answer key, as the actual accuracy of the subjects is of only minimal interest, but I can construct the "cultural" answer key, ie the set of answers that people consider highly deceptive or truthful.

Cultural Consensus Theory encompasses a type of statistical models that allow researchers to infer the shared cultural knowledge of informants [Batchelder et al., 2018]. Cultural knowledge refers to knowledge that is shared across a given group. For example, American restaurant-goers may share a cultural knowledge of how much money to leave as a gratuity in addition to their bill. This knowledge may be different for a different group, say a different country. Typical tests where Cultural Consensus Theory is applied have dichotomous answers, oftentimes restricted to True and False (Is this disease contagious? [Oravecz et al., 2012] Is this dog aggressive? Is this formula applied correctly?), although it is possible to modify the model to calculate for answers beyond the binary possibilities [Anders et al., 2014]. An informant that is presented with this task fills out the answers to the best of their knowledge. This informant may not answer every question correctly; the proportion correct will rise and fall with the informant's individual competence. However, when you assemble the answers of many informants together, you can use their combined knowledge to extrapolate the cultural answer key.

At its core, Cultural Consensus Theory is an application of signal detection theory. All informants have a chance to know the correct answer based on their competence, and if they do not know they will guess. Finding the correct answer among the noise and wrong answers is the signal I are trying to detect.

My model is a modified version of the model presented in Oravecz et al [Oravecz et al., 2014]. That model is set up to take only binary inputs in a True/False fashion. My model instead takes trinomial inputs corresponding to the three possible choices of the subject. I call this the multinomial extension of Bayesian Cultural Consensus. The model is coded in JAGS [Plummer et al., 2003] and executed through PyJAGS, a Python interface to JAGS [Miasko,

2017]. The code to the model is shown in Appendix C.

The data for the model $\boldsymbol{Y} = (Y_{ik})_{n\times m}$ is a n by m (where n is the number of subjects and m is the number of triplets) matrix consisting of values 1, 2, 3, or empty, with 1 representing the lie, 2 and 3 representing the true statements, and empty meaning this subject did not receive this triplet. The response cases are enumerated below. As a prior, we assume that the probability $p(Y_{ik})$ of a subject picking any answer is uniform, such that $p(Y_{ik} = 1) = p(Y_{ik} = 2) = p(Y_{ik} = 3) = 1/3$

$$Y_{ik} = \begin{cases} 1 \text{ if subject i responds with the lie to item k} \\ 2 \text{ if subject i responds with the first truth to item k} \\ 3 \text{ if subject i responds with the second truth to item k} \end{cases}$$

To get the probability of a subject i knowing the culturally correct answer (in our case, the lie) for question k, we need to find their specific competence, represented by D_{ik}. This value depends on the question difficulty δ and the subject's individual ability θ. The harder the question, the more likely it is that a subject will give an incorrect answer, and is represented by a higher value for δ, which lowers the value of D_{ik}. On the other hand, a higher individual ability θ increases the chance they will know the answer and consequently raises D_{ik}.

$$D_{ik} = \frac{\theta_i(1-\delta_k)}{\theta_i(1-\delta_k) + \delta_k(1-\theta_i)} \tag{4.1}$$

The probability distribution $p(Y_{ik})$of the subject's answers is thus determined by D_{ik}. The probability of the subject knowing the correct answer is D_{ik}. If the subject does not know the correct answer, the subject will guess with assumed uniformity from all three options. So the probability of a correct answer is D_{ik}, the probability of knowing the answer, plus $\frac{1-D_{ik}}{3}$, the probability of guessing the right answer randomly. The remaining options only

have the possibility of being chosen randomly, probability $\frac{1-D_{ik}}{3}$.

$$p(Y_{ik} = x) = \begin{cases} D_{ik} + \frac{1-D_{ik}}{3} & \text{if } x = 1 \\ \frac{1-D_{ik}}{3} & \text{if } x = 2 \\ \frac{1-D_{ik}}{3} & \text{if } x = 3 \end{cases} \tag{4.2}$$

The advantage of this model is that the ability to form a "cultural" answer key for all the triplets. It can do this even though not every subject sees every triplet, as the model can extrapolate from given data. As stated before, I am *not* interested in reconstructing the 'true' answer key from the answers of the Mechanical Turkers. What I am interested in is *bias*. When I sample from the model using a Markov Chain Monte Carlo sampler, I want to see what answers become highly selected, or 'dominant,' and which answers are rarely selected, or 'avoided.' This cultural answer key may not tell me what answers are truly deceptive, but it should tell me what the subjects *think* is deceptive.

I ran the model in six MCMC chains with 1000 samples each. This results in a matrix of sampled answers, with each column corresponding to the answers of a single triplet. By examining the distribution of answers in each column, I can see which answers were popular and which were avoided. I also generated 100 matrices of identical style and shape to the matrix of Turker answers, but that contained randomly chosen answers instead. I ran these randomly generated matrices through the model and compared them to the results of the Turkers to check what, if any, behaviors might be explained by random chance.

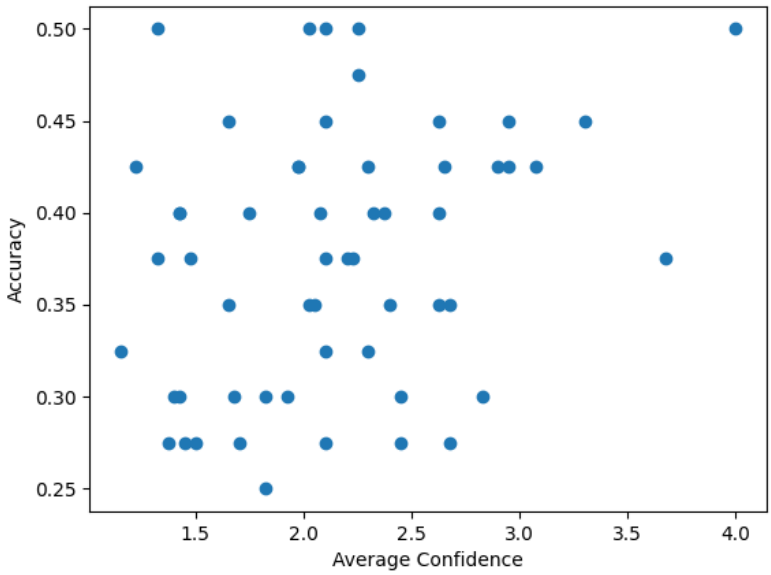

Figure 4.2: A scatter plot showing how accuracy changes with average confidence.

4.2 Results

After removing subjects that failed the attention checks, I obtained 2072 responses across the 300 triplets. The accuracy of the subjects is shown in Table 4.2. The accuracy is tracked in a variety of different metrics: the total accuracy across all subjects, the accuracy when binning by each confidence level (1-4), and by whether or not the subject left feedback on their response. Only when tracking Confidence 4 (Certain) and With Feedback do the responses achieve an accuracy above chance.

Looking at the user-generated rationales for their choices (ignoring tautological feedback such as "because it is a lie"), I find that most people look for logical inconsistencies or general plausibility when making their choice. In a select few cases there was a cultural knowledge failure. One respondent wrote that it was odd for the story to reference Lord of the Rings, as the player mentioned living in Middle Earth. This is not actually as implausible as it sounds: all the participants who created the original Motivated Deception Corpus were University of

Table 4.2: Judgement accuracy measure in total, by confidence level, and by feedback presence.

Grouping	Accuracy
Aggregate	32.5%
Confidence 1	30.05%
Confidence 2	33.7%
Confidence 3	32.2%
Confidence 4	37.8%
With Feedback	36.1%
With Feedback (tautologies removed)	29.8%
Without Feedback	32.2%

California, Irvine students. One of the student housing communities is called Middle Earth, with each dorm being a Lord of the Rings reference. However, since the Turker did not know this, it seemed suspicious to them and caused them to mark it false. Subjects also listed reasons such as vagueness in the story or grammatical errors as indicators of deception. This approach does not seems to serve people well, as this kind of response is linked to an accuracy of 29.8%, slightly below chance.

I also tracked the accuracy of each subject by their average confidence level. This plot is shown in Figure 4.2. The accuracy of a subject visibly trends upward with confidence, although with large amounts of noise.

After inputting the response matrix to the JAGS Bayesian Cultural Consensus model, I obtained the cultural answer key. This key was 31.3% accurate to the ground truth answer key, making the human subjects slightly worse than random chance in terms of accuracy. In order to determine the presence of a bias, I also tracked two metrics I refer to as *dominance* and *avoidance*. The dominance of a triplet is the proportion of times that its leading answer was picked out of all answers picked. For example, if option 1 of a triplet was picked 20 times, option 2 was picked 70 times, and option 3 was picked 10 times, option 2 would be the dominant option with 70% dominance. Stories with high dominance can be considered to seem more deceptive than their fellows. Avoidance uses the same process but in reverse;

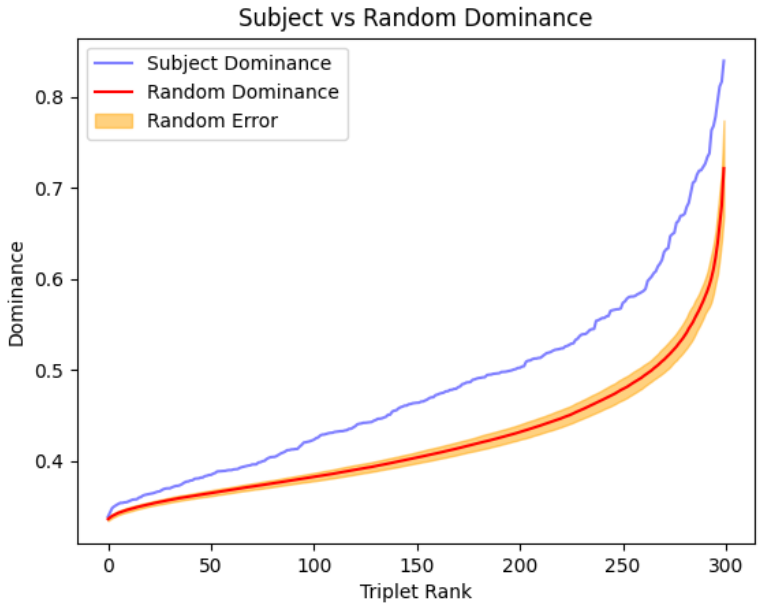

Figure 4.3: Subject dominance compared to random dominance.

the lowest option is considered avoided and can be inferred to be thought of as the most truthful of the triplet. In the previous example, option 3 would be the avoided answer at an avoidance of 10%.

The dominance and avoidance generated by the model is shown in Figures 4.3 and 4.4, along with a comparison to random data. The data is ordered in ascending and descending order respectively. The subject data is well outside of the random error, showing that the subjects decisions are not pure chance. There is some bias that is affecting which stories they view as deceptive or truthful.

These dominant and avoided stories became the input for my BERT model. This was a simple binary classification task to see if the network can determine the difference between the avoidant and dominant stories. The answer, generally, is no. After 500 training epochs, the highest test accuracy the model achieved was a mere 56%. This is the same network that achieved 93.6% on the gold-standard Ott dataset.

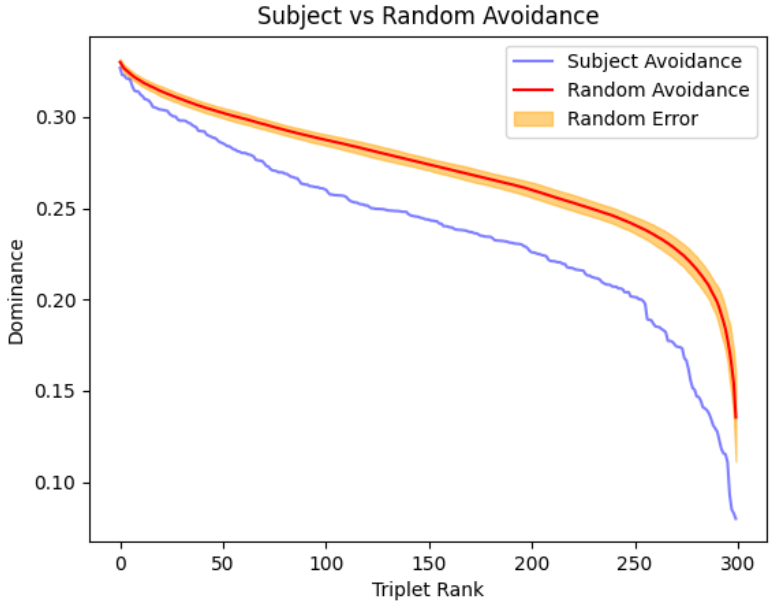

Figure 4.4: Subject avoidance compared to random avoidance.

Dominant Stories
My pet chinchilla once ate my homework.
Even though my family is paying for my college tuition, I plan to cut them off in the future, since they cause me a lot of mental issues. I have never once liked them.
When I was moving into a new house the moving people we hired tried to call the cops on us and I begged them not to.
A guy my roommate was seeing once saw me changing because they came into our room without telling me first.
I mostly do all of my shopping online using Amazon prime because of how convenient it is. I usually wait until Thanksgiving or Christmas to buy things due their sales.
I dropped my homework in a duck pond a few years ago

I haven't had a gf since middle school. I get matches on Tinder but as soon as the conversation starts I get ghosted.
I have never made any impulsive decisions when I was shopping.
In my apartment, my roommates and I found the biggest rat we had ever seen in our bathroom. Evidently so, we locked ourselves in the bathroom with the rat, as we were trying to capture it. That clearly didn't work as the rat ended up crawling into a hole near the corner of our bathroom and we lived in fear for the rest of the year.
I love shopping on amazon it's one of the only things that gives me serotonin anymore

Table 4.3: Ten stories that achieved more than 60% dominance in the model. Entries have been reproduced verbatim. The full list is viewable in the Appendix.

In terms of makeup, these dominant and avoidant stories are very similar, although there are a few trends that can be gleaned. My first examination is of the part-of-speech types of words that appear in the stories. I used the Natural Language ToolKit's part-of-speech tagger to determine how many of a given part of speech appears in each story [Loper and Bird, 2002]. As a measure of raw proportions, I cannot show that any particular part of speech is definitively more prevalent in one case than the other; there is simply too much variability. Avoided stories are slightly longer than dominant stories, 32.35 tokens on average compared to 27.97, but the there is also strong variability and the difference is weak overall. That does not mean, however, that there is no difference in how they are used.

Avoided Stories

I love going to the ARC. I've used both weight rooms on the right side and also towards the back. There are many amenities and I'm glad that we get them for "free" as a student.
I once went all the way to Toronto to see my favorite artist perform his first ever stadium concert. It was one of the best experiences of my life and I will never forget it.
I was pickpocketed twice when I visited New York.
I went on a trip to Italy and Greece for my 20th birthday!
I eat dinner everyday because I believe that proper nutrition is important especially when working out.
One time I told my friend we should make dinner together, so we went to my apartment to cook and after we spent some time talking he told me he had a crush on me and wanted to ask me out. I had a boyfriend.
I have a friend who loves to eat pasta everyday for dinner. I am the opposite and I can't stand having it often.
My first date was at the movie premier to Toy Story 2.
I have been consistently running/going for walks 3x a week during the past month.

Table 4.4: Ten of the stories that achieved less than 15% avoidance in the model. The full list is viewable in the Appendix. Entries have been reproduced verbatim.

In examining the dominant and avoided stories, I decided to limit the scope to what I call the exemplar stories– that is, the 38 stories with a dominance greater than 60% and the 20 stories with an avoidance lower than 15%. These limits were decided arbitrarily, in order to exclude stories that did not seem to make much of an impression on the Turkers. These

exemplars then, are stories that are strongly perceived as being either truthful or deceptive. Again, this is not the same as actually *being* truthful or deceptive. 60% of the avoided exemplars were indeed truthful, while only 28.9% of the dominant exemplars were actually deceptive.

When examining the more consistently selected stories, the exemplars, certain words appear more often in one category than the other. Words that are extreme or negative, such as *no*, *never*, *not*, *n't* (as in the suffix of hadn't or shouldn't), and *actually* appear more often in dominant stories than avoided ones, implying that people tend to find these terms deceptive. The evidence for passes a confidence interval of 90%, but many of the differences between the two categories are extremely subtle. You can view some of the consistently dominant and avoided stories in Tables 4.3 and 4.4.

More than simply the amount of words used though, I decided to look at how they were used. Collocations between words happen when two or more words are juxtaposed with each other with a frequency greater than chance. For example, the words 'bright idea' may be collocated in a document because these words often appear together as a bigram, but the words 'tiny table' might not, since although the words may appear together at some point, they do not do so frequently. Collocation is typically tracked with two metrics, the log likelihood ratio, which describes how frequent the collocation is, and the mutual information, which compares the probability of finding the words together to the probability that the words are independent.

The equations for log likelihood λ depend on several variables. The variables c_1, c_2, and c_{12} represent the frequency in the corpus of word 1, word 2, and the bigram of words 1 and 2 respectively. N is the number of total words in the corpus. p, p_1, and p_2 represent probabilities, calculated as $p = c_2/N$, $p_1 = c_{12}/c_1$, and $p_2 = (c_2 - c_{12})/(N - c_1)$. The variables are used in the following equation:

$$\log\lambda = \log L(c_{12}, c_1, p) + \log L(c_2 - c_{12}, N - c_1, p) - \log L(c_{12}, c_1, p_1) - \log L(c_2 - c_{12}, N - c_1, p_2) \tag{4.3}$$

$$L(k, n, x) = x^k(1 - x)^{n-k} \tag{4.4}$$

The equation to find mutual information is simpler, as shown in equation 4.5, depending only on the probability of one word P(x), the probability of the second word P(y), and the probability of the bigram of the two words P(xy). It's important to note that while I use mutual information as a measure of relative significance, it's important to keep the likelihood in mind, since the equation for mutual information can give anomalously high values for rare events. Thus it is best to check mutual information of bigrams that appear with a certain minimum frequency. I use the tools in the Natural Language ToolKit and the corpus analysis software AntConc [Anthony, 2004] to track both.

$$\text{MI(x,y)} = \log_2 \frac{P(xy)}{P(x)P(y)} \tag{4.5}$$

There are notable differences in the kinds of collocations that appear in the dominant and avoided stories. Since all the stories are told in the first-person perspective, I track collocations with the word 'I." There is a strong difference when it comes to the bigram 'I was.' It is much more prevalent in the dominant stories, while barely being present in avoided stories. To check this, I compared the incidence of the 'I was' bigram to what might be expected in a

Table 4.5: Strongest Collocations of 'I' between dominant and avoidant stories with their log likelihood (LL) and mutual information (MI), sorted by log likelihood. Collocates shown are those generated by AntConc that passed a p-value threshold of 0.05

Dominant			Avoidant		
Collocate	LL	MI	Collocate	LL	MI
was	36.848	3.141	have	21.835	3.269
have	27.022	3.415	can	11.874	3.439
had	14.377	3.929	love	10.779	3.854
get	12.1061	3.514	will	10.779	3.854
dropped	10.953	3.929	am	10.779	3.854
haven	10.953	3.929	lived	10.779	3.854
ve	10.953	3.929	–	–	–

set of randomly selected stories. The dominant stories have a higher value than what can be expected from chance ($p<.05$). In addition, when looking at the list of bigrams with 'I' that have the highest mutual information, the bigrams from the dominant stories have a higher proportion of words in the past tense like 'was' or 'had' than bigrams from the avoidant stories, which also contains words like 'will,' 'love,' or 'think.'

Following this, I checked for the presence of narratives in the stories. Stories in the Two Truths and a Lie corpus tend to come in two broad categories: narrative stories that relate tales about the writer's past and factual stories that tell about some attribute of the writer. There are also borderline cases where writers relate facts about their past. I tagged every story in the exemplar categories with it's status as either narrative, factual, or borderline. I found that 56.5% of the dominant stories were narrative compared to 35% of the avoidant stories. This disparity passes a 92% confidence interval that the two categories have different amounts of narrative.

As a last experiment, I conducted an informal survey where I asked respondants to identify any linguistic differences between the list of highly dominant and highly avoidant stories (labeled as list 1 and list 2 for the purposes of blind testing). 34 percent of the respondants identified the avoidant stories as being longer, despite the minor difference in tokens per sentence. However, 62 percent noted that the dominant stories had worse English than the

avoided stories, often pointing to sentence fragments and simpler sentence structure. A few respondants also mentioned that the avoided stories tended to be more positive in tone than the dominant stories.

4.3 Discussion

The results of my Bayesian Cultural Consensus model clearly shows that subjects have a bias when identifying deception that cannot be explained by random chance. This supports the premise that humans consider certain aspects of text deceptive or truthful. Identifying those exact patterns however is a far more difficult task.

Levine et al posits the idea that humans will by default assume something is true unless they have a reason not to [Levine, 2014]. My investigations support this result. Results like the mutual information in the bigram 'I was' resemble random chance for the avoided or perceived true stories but have unusual prevalence in dominant or perceived deception stories. This can indicate that the 'I was' bigram is symptomatic of a quality that causes humans to believe the story is made-up.

Note that whatever conclusions I draw here are not hard and fast rules. People are different, and what one person may consider deceptive may not be the same for someone else. Instead I am looking for trends, broad patterns that cause a piece of text to be more often perceived as truthful or deceptive.

The prevalence of the 'I was' bigram pointed me to a larger pattern. Terms like 'I was,' along with the increased presence of past tense words in the collocations, indicate a certain style of text even though the actual number of past tense verbs between the categories is broadly the same. What these kinds of words indicate are *narratives*, actual sequences of events that relate in some way to the writer. I tested this hypothesis and found the dominant stories

have a noticeably higher proportion of narrative stories than the avoidant stories. This leads me to believe that people tend to believe facts presented *fait accompli* more often than they believe a personal narrative. It's the difference between someone saying, "I'm a professional surfer with six gold medals" versus "I was in Hawaii for vacation and decided to do get paid for doing that."

The logical conclusion is that if you wish to be believed through text, stating simple facts rather than an involved story is more likely to be trusted. This is surely not the only phenomenon that humans use to decide what is trustworthy, but the impact of a narrative is difficult to deny.

Chapter 5

Conclusion

This research demonstrates how we can use deep learning and corpus techniques to improve our understanding of deceptive text. In this dissertation, I've approached the issue of unraveling text-based deception from multiple angles. The work described here leverages both the power of neural networks and realistic human data to identify patterns and provide new, high-quality data on deception. In the previous chapters, I studied deceptive text through three interrelated projects: (I) the construction of a state-of-the-art classifier, and using the response of that classifier to variations in the input to determine what parts of that input are informative, (II) creating an entirely new corpus of deceptive text that utilizes game structures to produce realistic samples of deception, and (III) a human subject study through Amazon Mechanical Turk that sheds light not on what makes a piece of text deceptive, but what makes a piece of text *seem* deceptive.

We know that patterns exist within deceptive text. Our high-performing BERT-based classifier was able to not only achieve a high level of accuracy on a gold-standard dataset, but by ablating specific parts of the input I was able to see not only that certain word types like singular nouns were informative, but that the critical "swing sentences" had differences

in their variability between truth and deception. I even built a deception generator of our own designed to produce samples of pure deception or truth. These samples might not be studied as any great literature, but they reinforce the idea that there are characteristics that mark deceptive text as being untruthful, and perhaps not the other way around.

To dig deeper into these characteristics, I needed larger, better data. In an attempt to obtain the highest-quality, most representative deceptive text possible, I made an entirely new corpus using a novel technique. This corpus, the Motivated Deception Corpus, was made not through a survey or scraping forum posts, but by gathering users and having them play a game where they were rewarded both for making good deceptions and seeing through the deceptions of others. Because of this, we now have a large, high-quality dataset to use in our experiments and those of any researcher working in this field. This data is deceptive in the best sense of the word; our mighty classifier that performed so highly on the previous gold-standard struggled. It was better than the human performance, but not by much, except in one area. When it was given the context of another story, the classifier accuracy improved significantly. It is safe to say that this corpus will provide a challenge to any researcher who wishes to test their theories.

It was this very dataset that I presented to the industrious workers at Amazon Mechanical Turk, so that I could determine how humans made their judgements. I was able to leverage Cultural Consensus Theory to calculate the cultural knowledge, or in this case perceptions, of the workers when it came to what made text deceptive. This is an extremely difficult task, as the results can be highly varied and people themselves cannot necessarily articulate what makes them pick a certain piece of text. While their accuracy was no better than chance, I was able to glean something interesting: the presence of a narrative in the text makes it more likely that it will be perceived as deceptive.

Studying deception is a long, ever-changing prospect that will likely never truly be finished. That does not mean, however, that it is impossible to make progress. The tools that I created

here in these projects are not only invaluable to our experiments, but to any experiments in this field. Our conclusions here are themselves tools for those who wish to become aware of what deceptive text might truly look like, what they might unconsciously label as deceptive, or what they might do if they want to make a piece of text that is more likely to be believed. Utilizing the fruits of these projects will bring us closer to a more complete understanding of deceptive text.

Bibliography

M. Abouelenien, V. Pérez-Rosas, R. Mihalcea, and M. Burzo. Detecting deceptive behavior via integration of discriminative features from multiple modalities. *IEEE Transactions on Information Forensics and Security*, 12(5):1042–1055, 2016.

Amazon. Amazon customer reviews dataset, 2014.

R. Anders, Z. Oravecz, and W. H. Batchelder. Cultural consensus theory for continuous responses: A latent appraisal model for information pooling. *Journal of Mathematical Psychology*, 61:1–13, 2014.

L. Anthony. Antconc: A learner and classroom friendly, multi-platform corpus analysis toolkit. *proceedings of IWLeL*, pages 7–13, 2004.

S. Aphiwongsophon and P. Chongstitvatana. Detecting fake news with machine learning method. In *2018 15th international conference on electrical engineering/electronics, computer, telecommunications and information technology (ECTI-CON)*, pages 528–531. IEEE, 2018.

D. Barsever, S. Singh, and E. Neftci. Building a better lie detector with bert: The difference between truth and lies. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2020.

D. Barsever, M. Steyvers, and E. Neftci. Building and benchmarking the motivated deception corpus: Improving the quality of deceptive text through gaming. *unpublished*, est 2022.

W. H. Batchelder, R. Anders, and Z. Oravecz. Cultural consensus theory. *Stevens' handbook of experimental psychology and cognitive neuroscience*, 5:201–264, 2018.

V. Carrieri, L. Madio, and F. Principe. Vaccine hesitancy and (fake) news: Quasi-experimental evidence from italy. *Health economics*, 28(11):1377–1382, 2019.

D. L. Chen, M. Schonger, and C. Wickens. otree—an open-source platform for laboratory, online, and field experiments. *Journal of Behavioral and Experimental Finance*, 9:88–97, 2016.

N. R. Council et al. *The polygraph and lie detection*. National Academies Press, 2003.

D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. A practical part-of-speech tagger. In *Third conference on applied natural language processing*, pages 133–140, 1992.

B. de Ruiter and G. Kachergis. The mafiascum dataset: A large text corpus for deception detection. *arXiv preprint arXiv:1811.07851*, 2018.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

S. Feng, R. Banerjee, and Y. Choi. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 171–175. Association for Computational Linguistics, 2012.

E. Filatova. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Lrec*, pages 392–398. Citeseer, 2012.

E. Fitzpatrick, J. Bachenko, and T. Fornaciari. Automatic detection of verbal deception. *Synthesis Lectures on Human Language Technologies*, 8(3):1–119, 2015.

E. B. Ford. Lie detection: Historical, neuropsychiatric and legal dimensions. *International Journal of Law and Psychiatry*, 29(3):159–177, 2006.

T. Fornaciari and M. Poesio. Identifying fake amazon reviews as learning from crowds. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 279–287. Association for Computational Linguistics, 2014.

T. Fornaciari, L. Cagnina, P. Rosso, and M. Poesio. Fake opinion detection: how similar are crowdsourced datasets to real data? *Language Resources and Evaluation*, 54(4):1019–1058, 2020.

J. Frijters, A. Kooistra, and P. Vereijken. Tables of d for the triangular method and the 3-afc signal detection procedure. *Perception & psychophysics*, 27(2):176–178, 1980.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

T. Highfield. *Social media and everyday politics*. John Wiley & Sons, 2017.

D. Hovy. The enemy in your own camp: How well can we detect statistically-generated fake reviews–an adversarial study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 351–356, 2016.

S. Hu. Detecting concealed information in text and speech. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 402–412, 2019.

D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits. Is bert really robust? natural language attack on text classification and entailment. *arXiv preprint arXiv:1907.11932*, 2019.

B. Kalsnes. Fake news. In *Oxford Research Encyclopedia of Communication*. 2018.

M. Khodak, N. Saunshi, and K. Vodrahalli. A large self-annotated corpus for sarcasm. *arXiv preprint arXiv:1704.05579*, 2017.

R. Kuhn and R. De Mori. A cache-based natural language model for speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 12(6):570–583, 1990.

T. R. Levine. Truth-default theory (tdt) a theory of human deception and deception detection. *Journal of Language and Social Psychology*, 33(4):378–392, 2014.

J. Li, M. Ott, C. Cardie, and E. Hovy. Towards a general rule for identifying deceptive opinion spam. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1566–1576, 2014.

E. P. Lloyd, J. C. Deska, K. Hugenberg, A. R. McConnell, B. T. Humphrey, and J. W. Kunstman. Miami university deception detection database. *Behavior research methods*, 51(1):429–439, 2019.

B. Local. Local consumer review survey— online reviews statistics & trends, 2018.

E. Loper and S. Bird. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*, 2002.

Q. Mandjes. (in) capable deceivers: What a game of poker tells about possible individual deceiving differences. Master's thesis, University of Twente, 2019.

T. Miasko. pyjags (version 1.2. 2)[computer software], 2017.

R. Murphy. Local consumer review survey. *BrightLocal. Retrieved*, 19, 2018.

T. Niven and H.-Y. Kao. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*, 2019.

W. S. Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.

Z. Oravecz, J. Vandekerckhove, and W. H. Batchelder. User's guide to bayesian cultural consensus toolbox. 2012.

Z. Oravecz, J. Vandekerckhove, and W. H. Batchelder. Bayesian cultural consensus theory. *Field Methods*, 26(3):207–222, 2014.

M. Ott, Y. Choi, C. Cardie, and J. 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: Human language technologies-volume 1*, pages 309–319. Association for Computational Linguistics, 2011.

M. Ott, C. Cardie, and J. T. Hancock. Negative deceptive opinion spam. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 497–501, 2013.

A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10, pages 1320–1326, 2010.

G. Paolacci and J. Chandler. Inside the turk: Understanding mechanical turk as a participant pool. *Current directions in psychological science*, 23(3):184–188, 2014.

J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

M. Plummer et al. Jags: A program for analysis of bayesian graphical models using gibbs sampling. In *Proceedings of the 3rd international workshop on distributed statistical computing*, volume 124, pages 1–10. Vienna, Austria., 2003.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*, 2018.

Y. Ren and D. Ji. Neural networks for deceptive opinion spam detection: An empirical study. *Information Sciences*, 385:213–224, 2017.

P. Shrestha, S. Sierra, F. A. González, M. Montes-y Gómez, P. Rosso, and T. Solorio. Convolutional neural networks for authorship attribution of short texts. In *EACL (2)*, pages 669–674, 2017.

K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36, 2017.

I. Sutskever, J. Martens, and G. E. Hinton. Generating text with recurrent neural networks. In *ICML*, 2011.

D. Tang, B. Qin, and T. Liu. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1422–1432, 2015.

S. Tasnim, M. M. Hossain, and H. Mazumder. Impact of rumors and misinformation on covid-19 in social media. *Journal of preventive medicine and public health*, 53(3):171–174, 2020.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

N. Vogler and L. Pearl. Using linguistically-defined specific details to detect deception across domains. *Natural Language Engineering*, 1(1):1–32, 2019.

A. Wang and K. Cho. Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*, 2019.

W. Y. Wang. " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*, 2017.

S. C. Weller. Cultural consensus theory: Applications and frequently asked questions. *Field methods*, 19(4):339–368, 2007.

T. Wolf, J. Chaumond, L. Debut, V. Sanh, C. Delangue, A. Moi, P. Cistac, M. Funtowicz, J. Davison, S. Shleifer, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, 2020.

Q. Xu and H. Zhao. Using deep linguistic features for finding deceptive opinion spam. *Proceedings of COLING 2012: Posters*, pages 1341–1350, 2012.

F. Zamora-Martinez, V. Frinken, S. España-Boquera, M. J. Castro-Bleda, A. Fischer, and H. Bunke. Neural network language models for off-line handwriting recognition. *Pattern Recognition*, 47(4):1642–1652, 2014.

Appendix A

Appendix A: Truthful Swing Sentence

Truthful Swing Sentences
recently returned from a two - night stay at ambassador east hotel.
the recently remodeled affina was amazing - from the 6 choice pillow menu to the stocked ” pantry and refrigerator ” to the brand name bathroom ammenities.
once in a while you come across hotels with great service, well thought out rooms in a perfect location (right off magnificent mile).
we could easily walk to the red line or access the bus lines along michigan avenue.
a bunch of us got together and we had a great time in this hotel we asked for limes and they gave us like a punch bowl of them the rooms were so awesome you really have to see it to believe how extradionary this hotel is i love the decorations on every floor and being surrounded by such elegance.

as a royal ambassador member, they upgraded me to a beautiful junior suite with a separate living and working area and 2 bathrooms!
ideal position, lovely quiet rooms, good facilities, complimentary breakfast well received and the manager's evening drinks reception excellent ; we always tipped the staff who were serving our drinks.
as we were diamond members we got upgraded to a suite which was great because we were provided access to their executive lounge (free food & drinks).
beautifully appointed, professionally staffed, comfortable, well - located, and elegant, with wireless internet access (plus free access in an alcove on the second floor), fitness center (a little small but fine if you use it at the right time), good energy clientele, easy access to michigan avenue, next door to 24 - hour restaurant, 2 blocks from 2 starbucks, elevators that are there before you get your finger off the button - they thought of everything.
there are other good choices in town, but nobody has what the james offers : superior service - aka class meets efficiency -, metropolitan design and comfort with an attention to details i found only at the w in seoul, and what always plays a role meaning a convenient location... then they get to you with small unexpected touches that make the difference ie : i was about to take off and drive north, it was a hot july day... one of the staff members handed me a bottle of water.

when i checked in, the staff was very attentive, i showed a little consideration for the friendly attitude (small tip) and was upgraded to a huge upper floor room complete with a bar!
the kids loved the goldfish in the room and thought the window seats were the best.
walking distance from the art museum, millennium park, grant park (right across the street) and a quick cab ride to mccormick place.
we had our hotel reservations at another hotel set and after we were reading all of the negative reviews we cancelled and made out reservation at the hilton - i am so glad we did the hotel was clean, hotel staff was pleasant and helpful and the beds were amazing.
i landed the hotel through priceline for 80*pernightandpaidanadditional* 75 per night for a suite on the 41st floor.
it is in a great location - walking distance to millenium park, the loop and michigan ave.
very accomodating.. had 2 dble beds and bathroom was fine.... a bit small compared to usa hotels.... but compared to european travel... the place was palatial!
we had no problems at this hotel... rooms, service, location were top notch.

all rooms are suites, staff is friendly and professional, restaurant is out of this world and all the extras like a great health club, turn down service, complimentary morning beverages delivered to room and on and on.
the location is superb too, attached to nordstrom and north-bridge mall.
the suite was very quiet even though we were on the east side of the hotel, very near to michigan avenue.
white luxurous bedding on the king bed, plasma t. v. / subroofer / surround sound speakers (my boyfriend was more impressed than me, as i reminded him that we didn't come to chicago for the hotel room) very large bathroom w / tub and separate stand - in shower, granite countertops, his / her robes and slippers.
our suite has view on michigan avenue with a bit of the lake at the end.
this is the kind of rate structure one might expect from the 5 star hotels within spitting distance of the talbott, but not this less - than - maintained lodging.
the hard rock had a number of issues including : very dark room - not much light (artificial or natural) ; lots of street noise ; very cheap decor and furniture ; broken cabinet door ; very warm room - said they would send maintenance, but they never arrived ; had bar charges on account that were not ours.
prior reviews led us to believe this was a quality hotel ; not true.

the morning i checked out, our elevator stopped at every single floor from the 16th floor down to the lobby, but the elevator was almost completely full when i got on at the 18th floor, so i can only imagine how long it must have taken other guests to get downstairs.
here are my experiences : - had three rooms reserved, when checked in we were charged 2 different rates for the rooms - screwed up initial room assignments, gave us a room that was not even cleaned yet when we checked in - could not get us new keys to a different room without great difficulty (we had to go to the front desk because bell staff went to the wrong room to give us new keys) - in room guide says pool in on the 8th floor and signs say it is on the 7th - could not get to the pool from the 14th floor without going all the way down to the lobby and then had to go to the 7th floor and take the stairs to the 8th - when we checked in we had the front desk staff go over which credit cards were assigned to which room.
certaintly the location is great and the swimming pool is really nice and the room was generally well appointed oh and the bed was great.
she was short, rude, handed me my keys and walked away.
i've since been on the phone with a manager, who promised to call me back once she researched the incident, and have yet to hear back, despite the promise that she would call me back two days ago.

we might have had a one in a million experience, but we : - were served a hairy salad (a long black hair held a lettuce leaf hostage) ; - considered our guest room tiny ; - could not submerge a thigh, much less a body, in the tub (we're not obese and we're not dexter) ; - endured a broken air conditioner for days (it couldn't be repaired) ; - experienced the delight of an inexplicably broken toilet (they fixed it that day).
if you like staying somewhere that the air in the rooms doesn't go down past 74, the closet is 24 " wide, all old / dirty brass fixtures, beds that are a little bigger than a twin, showerheads that barely put out water, then this is your hotel.
compared to other hilton properties in chicago, the rooms were not as well appointed, the bathrooms in the rooms were smaller and not as nice, the tv was tiny, the hallways on the executive floor were very noisy all night long, and the breakfast you get was more like a hampton inn than a 4 star hotel.
i called on december 6th, 2010 at 3 p. m. to book a room for an upcoming weekend getaway, and was taken aback when the phone was picked up and promptly hung up without a word.

pros - location by the chicago river, two blocks from michigan avenue - restful lobby with river view and free wifi provides a pleasant work environment - buffet breakfast offering custom omelettes, crepes and french toast cons - check - in assigned me to a city view room when i had prepaid for a river view room - river view was good, but the windows to see it were filthy - dirty towel used by the previous guest was still hanging on the back of the bathroom door - bathroom was unimpressive : bathtub - shower combination ; worn plumbing fixtures ; only 2 sets of towels provided - guest room was small : only one soft chair provided - guest room doors slammed noisily throughout the floor hallway - 7 to 8 block walk to nearest l - station - concierge couldn't explain how to get to the mercury theatre, nor print a map of the area - guest services told me to go out of the hotel one block to get starbucks coffee when it was available downstairs on the link cafe - no complimentary newspaper delivered to the room - no final billing folio delivered to the room on the night before check - out - not all computers were operational at the link cafe where boarding passes can be printed
our suite at the knickerbocker are quite nice, but the behavior and attitude of the staff is beyond abysmal.
not a tremendous problem, but when the young woman at the front desk replied " what do you want me to do about it " with the tone of a flippant teenager, i began to suspect that a severe deficiency in manners may have been evident in the staff.
fast forward to this evening.

following a lovely meal off the premises, we chose to turn in a bit early at 11 : 30 pm only to discover that despite our best intentions, the bass from a party downstairs was more than audible in our room.
acting prudently and with restraint, we decided to phone the front desk to inquire as to why our room had been inadvertently annexed into the party downstairs, but yet again we were met with a curt and quite rude response.
sean at the front desk informed us that the party was a greater priority for the hotel, and that the family dictated that the music was to be set at a certain volume.
$ 50 for valet.
it is 2010 - enough said.
even if you arrive in a cab, there is no space to load / unload so you have to block the street while you arrive / depart, the rooms seem a bit dated and in need of a refresh.
some of the walls had wall paper missing, most of the faucets leaked, there was water stains on the ceiling in the bathrooms, some of the furniture had very visible stains, the fire alarm was coming off the wall in the main bedroom.
through our 2 week stay, they told us to " have a nice day " as we left in the am and welcomed us back upon our return.
breakfast is served in the am, and the servers were as delightful as the bellmen.
she had booked the hotel but had added my name to the reservation since i would be arriving first.

the lobby of the homewood is right down the hall from the kitchen and dining room (actually more of a grade school cafeteria type set up).
from the reservations assistant (who took 45 minutes to book a room) to check out (they billed my card for 750 dollars) and getting a refund (numerous voicemails and emails no one returned my call).
the front of the hotel is block and lucky for us the protestors that are standing in front of the hotel were able to tell use where the new entrance was.
from the wrong food with room service, to room keys that dont work, unrefreshed bath towels, forgotten and dirty in - room coffee service, unclean ice bucket overnight, forgotten room cleaning, inability to connect to the internet.... i'm too tired to keep writing all the problems.
the house keeper only did the bare minimum in the room each day and did not replace the used toiletries or the coffee ; she also left the dirty coffee cups in the room.
we called the front desk and it took almost 40 minutes for someone to walk over to our room (we were on the 5th floor, same as the reception desk).
i was shocked and disapointed with my stay, here are some of my gripes - room had a broken phone and broken lightbulb - breakfast service was slow and impolite - hotel rationed shampoo (one bottle for two people sharing and did not refill a bottle that was less than half full).

the wireless was very slow and the hard - wire from the wall did not work at all.
needs more intense management.
3 phone calls to get maintenance to fix the bathroom sink, unpleasant experience in the terrace resturant, front desk not aware of room eminities, and a very uncaring mod!
: (
under the gloss of a nice building, friendly staff, and a wild set of items in the minibar, this is somewhat below - average hotel.
what about the slippers that, although being still in the bag, had all the sole broken and left tiny plastic parts over the way to the garbage (all over the room, de facto).
i then attempted to use both of the phones at the pool, one white phone and one emergency red phone, to call the desk.
as we were exited the pool area i ran into a hotel employee and told her about the problems and then asked her to call us when the pool was clean.... never heard back.
for example, he got locked out of our room 2 times, once having to wait for 15 minutes in the hallway after going to the spa.
it happened to me also, coming back from the spa, and i called on the hallway phone to get help.
luckily, right after that, an employee who was randomly walking by asked me if i had been waiting long - - as if she knew this was a problem.
it was a little after 10 : 30pm.

pros : elegant lobby nicely decorated with fresh flowers location : located in the heart of chicago disappointments - stained and worn out carpet in hallway - room was small - phone and light-bulb in room were broken - breakfast was $ 60 for two people, service was slow and impolite - hotel rationed soaps and sham-poo, one bar of soap to share and one bottle of shampoo which was not replaced for two days, i felt like i was staying at a budget hotel.

Table A.1: Truthful swing sentences as identified by our model, reconstructed from the tokenizer.

Appendix B

Appendix B: Deceptive Swing Sentence

Deceptive Swing Sentences
i loved staying at the hard rock hotel in chicago, not only is it an amazingly friendly atmosphere, but they give me the option to bring my pet with me.
the ambassador east hotel in chicago is a fantastic up - scale hotel to stay in while visiting the windy city.
the affinia chicago is a wonderful place to stay, my husband and i stayed there for a week to visit some family and had an amazing time.
the hard rock hotel chicago is great alternative to ordinary hotels.
upon arrival at the ambassador east hotel in chicago, i was immediately impressed with the courtesy and attentiveness of the staff.

the magnificent mile in chicago is a great place to visit, and staying at the affinia chicago just made it that much better!
elegant and luxurious with a beautiful ocean view.
my husband and i went over the holidays to see my family and we stayed at this hotel.
amalfi hotel chicago has several factors that make it one of the best hotels in the chicago area and an experience you will not forget in a long time.
the palmer house hilton in chicago is by far the best experience i have ever had away from home.
i will definitely stay here again next time in chicago.
i am so glad i decided to stay at the intercontinental chicago for my first trip to the city.
we really enjoyed our stay at the palmer house hilton.
my wife and me stayed in the amalfi hotel chicago last august in a weekend visit to chicago.
my fiancee and i were looking for a modern, upscale venue for our wedding reception.
the james chicago hotel is located right in the heart of the one and only, downtown chicago.
the chicago hilton is a great hotel our stay there was fantastic.
our stay at the hilton chicago was a pleasure from arrival to departure.
sofitel chicago water tower is a four star hotel that is minutes aways from the magnificent mile, navy peir, the museum of contemperary art, lake michigan and upscale boutiques.

we enjoyed our stay at the james in chicago very much.
i had a wonderful time at the james hotel while on business in chicago.
the hotel made my experience of visiting this city even more wonderful, i highly recommend this to anyone.
went to see the museum of contemporary art which was great, but this hotel almost had it beat!
my husband and i loved this hotel.
the homewood suites by hilton, in downtown chicago, has to be one of the most comfortable and affordable hotels in the windy city.
my stay at sheraton chicago hotel and towers was wonderful, i stayed in the traditional guest room and i slept good, i was able to get to my meeting well rested, thank you sheraton!
it has big rooms with comfortable atmosphere, huge bathroom. over all, i had the a great experience there, would definitely stay there again!
i stayed at swissotel chicago when i was on business and it was very nice.
this hotel is the best hotel ever in my opinion, and i really enjoy everything thing about it and i also have many different reasons why i like this hotel.
i will be staying there again due to the wonderful experience and decent prices.
the atmosphere at this hotel is truly remarkable ; it has a very modern feel.

the hyatt regency chicago hotel offered pda or kiosk check - in which was great.
the hyatt regency chicago is one of the most beautiful hotels that i have ever stayed at.
my girlfriends and i stayed at the hyatt in chicago during a shopping trip to the city in june 2010.
conrad chicago is one of the nicer hotels i have had the pleasure of staying in recently.
my husband and i stayed here for a short get - away weekend and loved it.
i had a great experience staying at the conrad chicago, the service was top notch, the price was good, but out of everything the service was outstanding.
i would definitely stay there again when in the area and would suggest it to anyone looking for a good quality hotel in a great location.
the ambassador east hotel is a terrible place to stay.
for 250 dollars, cheapest room available at the hard rock hotel chicago, you would assume you would have access to wifi.
for a hotel that touts its luxury location in chicago the talbot is very poor (think motel 6 at a hilton price).
the affinia in chicago obviously caters to wedding guests and corporations hosting business conferences.
my wife and i stayed at the abassador east hotel in august to attend the air and water show in chicago.
i was extremly unhappy with my recent stay at affinia chicago!

if you are looking for a high end hotel on the magnificent mile, the affinia chicago is not your best option.
terrible experience, i will not stay here again.
the hotel allegro located in the chicago loop, provided my wife and i with one of the worst hotel experiences in recent memory.
my wife and i were excited to visit and shop at chicago's magnificent mile on michigan ave. we chose the intercontinental hotel for it's reputation and location.
i was highly disappointed with my choice to stay at the amalfi hotel in chicago.
the international chicago magnificent mile is shrouded in glamour, but underneath the facade fades away.
my stay here was distasteful, and i never intend to return, except to ask for my money back.
my wife and i were guests at the hotel allego in chicago for a long weekend getaway (september 2 - 5, 2011).
when i went to the james hotel in chicago i truly expected a luxury experience exactly like what they were advertising.
having made regular business trips to chicago, i decided to stay at the hotel monaco chicago, since it was fairly new and i had a friend that had stayed there.
when we got checked and arrived at our room the first thing we noticed was the light didn't come on when we flipped the switch upon entering.
after arriving at the sofitel chicago water tower hotel i was greeted with rudeness and snubery.

i stayed at sofitel with my husband for a weekend and i will never be staying again!
what a sickening affair ; the hotel monaco chicago is a stuffy masterpiece of a hotel done wrong.
my wife and i recently stayed at the hotel monaco in chicago and, after our experience there, we will not be returning.
the hotel sofitel chicago water tower bills itself as a 4 - star luxury hotel, but the luxury is evidently in the eye of the beholder.
my husband and i arrived at the swissotel chicago to celebrate our 13th wedding anniversary.
i want everyone to know about the awful experience i had at the sheraton chicago hotel and towers.
i believe they canceled our reservation just so they could make more money.
i for one won't be going back to this hotel!
the sheraton chicago hotel and towers is a nice place to stay if you need a place to stay on quick notice, but it certainly does not " exceed expectations " as touted on their website.
i was really expecting a lot more from a quality chain like the sheraton, but a recent stay at their downtown location in chicago was somewhat of a disappointment.
this might be a classy looking hotel in downtown chicago, and while the location is not bad, the rooms are definitely hit or miss.
i really would expect great service for the high price of a stay here.

if i were to return to chicago again, i'd definitely try out staying somewhere else.
for the amount of money per night that the millennium knickerbocker hotel charges, one would at least expect a room with a working bathroom.
all in all, i was disappointed with my stay.
i recently stayed at the homewood suites by hilton chicago downtown, and found it lacking in the quality of their service.
the millennium knickerbocker hotel has seen better days.
i had fond memories of staying there when it was considered the finest hotel.
there are plenty of other places to spend your money.
don't leave it at this hotel.
i recently had the misfortune of staying at the swissotel chicago hotel in illinois, and it was one of the worse travel experiences i have ever had!
cleanliness seems to be an issue with the maid staff.
room service was adequate but not great.
i stayed at the sheraton chicago hotel and towers in december of last year and had one of my worst hotel experiences ever.
after a my stay at the millennium knickerbocker hotel chicago i contemplated the the poor quality of my experienc there and felt the need to document it.
our vacation was highlighted by a four day stay at homewood suites by hilton chicago downtown. i believed it would be the best part of the trip ; i discovered it was an unfolding nightmare.

when i walked into the millennium knickerbocker hotel, my first thought was, " wow this is very yellow ", and not the good kind of yellow like a sunny day, no the ugly yellow that you would want to wash off your pretty white blouse.
the swissotel chicago hotel aspires to be a tourist's paradise, a hotel so grand and luxurious that you'd rather stay than return home at the end of your trip.
the sheraton chicago is not the place to be if you want to experience chicago.
the room was very beautiful besides the faint mildew smell in the room.
room service was great and very pleasing! our business meeting took place in the st gallen room where our meeting took a little disruptive turn, there was construction going on during that time and was very hard to hear the announcer in our meeting.
the swissotel chicago is a very mediocre hotel, the service is always poor, and the room service food always comes cold, unless it's supposed to be cold than it comes warm.
i stayed at the sheraton chicago hotel for a business meeting and had an absolutely horrible experience.
i get up to my room and unload all of my bags only to find my a / c doesn't work.
i have never received such poor customer service in my life and will not be comming back.
while the hotel certainly seems to look beautiful, the hotel is actually far from it.

in general coming here is a bad decision despite how it looks, its a mistake i wont make again and you shouldn't either.
i was recently a guest at the sheraton chicago hotel and towers and was immensely dissatisfied.
i arrived to find that they had " lost " my reservation.
after nearly an hour of arguing with the front desk clerk she finally asked for the hotel manager to step in.
i had asked for a suite as i was going to be staying in the hotel for several days.
unfortunately, i was in town for a conference and everyone in the area was completely booked.
i had no choice but to accept what they had offered.
i begrudgingly accepted this with the intention of contacting the corporate office and posting my review here.
do not stay in this hotel they will not help you with anything, even if it was their error.
i stayed three nights recently at the hilton homewood suites - downtown chicago.
i stayed in the homewood suites in downtown chicago last week, and i was extremely disappointed with my experience.
i was sorely disappointed with the sheraton chicago.
we stayed in the millennium knickerbocker hotel chicago in a standard guest room.
i was very disappointed with this hotel.

room service took forever to pick up (good sign in a way = busy because they are good), but the food arrived very late and very cold.
overpriced is the best word to describe the conrad chicago hotel.
while it may be in downtown chicago, the room had no view.
although i asked for non - smoking, the room reeked of smoke.
there was a stain on the pillow, and the leg of one of the chairs in the room was broken.
there was a dead bug in the bath tub.
i was only given 2 towels for my 4 night stay.
i don't recommend this hotel at all!
hyatt regency hotel : good ole downtown, chicago.
well, when we got to our room, one of the first things my wife did was sneeze four times.
' hyatt regency chicago'has ruined our family holiday weekend.
beware of this place, if you want to enjoy your weekend in chicago, hyatt regency should be the last place you should consider staying.
recently staying at the omni chicago hotel, was a waste of money.
the hyatt regency hotel in chicago was the worst hotel i ever went to!
i was really looking forward to a nice relaxing stay at the end of a long vacation, but unfortunately that was not to be had.
i expected a bit more from the hyatt regency chicago i stayed at recently.

if you want a 5 - star hotel with 1 - star service, make sure you book your next chicago stay at fairmont chicago millennium park.
when i checked in to the hotel, i had to wait at the desk for over ten minutes with no line.
i was greeted with no " thank you for your patience, " or anything similar.
the room was fine, but nothing too glamorous or outstanding.
the internet cost $ 14. 95 / day which is outrageous considering the price of the hotel itself.
the thing that pushed me over the edge, however, was the valet parking.
when i went to complain at the front desk, they just said that sometimes the valets get overwhelmed, and you have to exercise patience!
i will exercise patience as i check out of this hotel!!
!
my husband and i recently stayed at the conrad chicago for three nights, a thursday through saturday.
hyatt regency chicago seemed like a nice place to stay, but we didn't like it very much.
the hyatt regency chicago basically caters to guests who want to feel like they're staying in a nice hotel as soon as they enter the lobby.
this hotel was full of drunks.

i wasn't aware of the noise level when i booked the room and when i threatened to make other reservations, the people in the lobby had nothing to say.
we asked for a water view and got a city view.
overall, we had a horrible experience.
when my boyfriend and i checked into our room at the omni chicago, we were irritated to see that the bed looked rumpled, as if kids had been jumping on it.
i would have been more comfortable staying in a run - down motel.
imagine flying to the windy city for business.
you've been expecting to fall into a soft bed with a lovely view of the chicago skyline out your window.
that's exactly how i imagined it after my delayed flight and terrible cab ride.
i get really uncomfortable when i am in a crowded area, and people are smoking their lungs out.
on a recent business trip to chicago, i had the unfortunate circumstance of staying at the fairmont chicago millennium park hotel.
the fairmont chicago millennium park hotel was supposed to be our romantic get away and it turned into a nightmare!
my stay at the fairmont chicago millennium hotel was a short one ; although, it was intended to be for five days, it only last two.
i went up to my room and looked around, all seemed ok at first.

Table B.1: Deceptive swing sentences as identified by our model, reconstructed from the tokenizer.

Appendix C

Appendix C: JAGS Code

```
model{
    for (i in 1:n) { # for each subject
        for (k in 1:m) { # for each question
           # Probability correct for subject i on question k
           D[i, k] <-
               (theta[i]*(1-delta[k]))/(theta[i]*(1-delta[k])+(1-theta[i])*delta[k])

           # Create a vector of probabilities of picking each response option
           for (j in 1:l) { # for each response option, l normally set to 3
              # Probability respondent picks answer j
              pYm[i,k,j] <- ifelse( z[k]==j , D[i,k], (1-D[i,k]) / (l-1))
           }

           Y[i,k] ~ dcat( pYm[ i,k,1:l ] )
        }
    }
```

```
    for (i in 1:n) {
        theta[i] ~ dunif(0,1) # individual ability
    }

    for (k in 1:m) {
        for (j in 1:3){
          iniv[k,j] <- 1/3}
        z[k] ~ dcat( iniv[k,1:3] ) }
    }

    delta[1]<- 0.5 # item difficulty
    for (k in 2:m){
        delta[k] <- tempdelta[k-1]
        tempdelta[k-1] ~ dunif(0,1)
    }
}
```

Appendix D

Appendix D: Dominant Stories

Dominant Stories
My pet chinchilla once ate my homework.
Even though my family is paying for my college tuition, I plan to cut them off in the future, since they cause me a lot of mental issues. I have never once liked them.
When I was moving into a new house the moving people we hired tried to call the cops on us and I begged them not to.
A guy my roommate was seeing once saw me changing because they came into our room without telling me first.
I mostly do all of my shopping online using Amazon prime because of how convenient it is. I usually wait until Thanksgiving or Christmas to buy things due their sales.
I dropped my homework in a duck pond a few years ago
I haven't had a gf since middle school. I get matches on Tinder but as soon as the conversation starts I get ghosted.
I have never made any impulsive decisions when I was shopping.

In my apartment, my roommates and I found the biggest rat we had ever seen in our bathroom. Evidently so, we locked ourselves in the bathroom with the rat, as we were trying to capture it. That clearly didn't work as the rat ended up crawling into a hole near the corner of our bathroom and we lived in fear for the rest of the year.
I love shopping on amazon it's one of the only things that gives me serotonin anymore
i lived in middle earth my freshman year
In the fourth grade, I had not done a homework assignment. Due to that my teacher decided to yell at me and make fun of me in front of the class.
For six years, since I moved to California from Texas, our furniture would not fit in the house since Texas houses were much bigger than Californian. So my family decided to keep the pool table and stick it in our living room rather than have a dining table to eat dinner on.
i went on a date with a guy that I had met online. When he had discovered I had never watched Star Wars. He told me that I could walk myself home.
I never really do exercise ever. Even though I have a planet fitness membership.
In elementary schools, I have many friends around 9 before third grade. But after I become part of the fourth graders, I only have one friend to play with.

I have actually been a victim of plagiarism. In high school, we were asked to write creative short stories on the topic of a Shakespeare play. We were then asked to bring in these drafts so we could do peer reviews with our classmates. I was put in a group of three, and one of the two girls, after reviewing my short story, copied one of my main plot events directly into her story. She actually delivered the twist in a sentence that sounded nearly identical. I was very upset and went to the instructor about it. She failed the assignment.
I applied for a double room in Vista Del Norte this year with my friends from high schools but failed to be on the list.
I once farted out loud during an exam on accident.
i walk a 5k each morning
I get major test anxiety. I begin to hyperventilate and get huge memory blanks as soon as it gets close to examination time. I have no memory of taking my Anthropology Exam first year.
When I was dorming in a quad room, my roommates often turned off the lights in bathroom when I was in the stall thinking that no one was in there. Then they left before I could tell them I was inside.
This year I am living in ACC housing. However, one day I woke up to find police outside the door. It turns out someone had died in a neighboring apartment.
Growing up as children of immigrants who started from zero, I was lucky enough to consistently live in new houses. I have moved and resided and 3 new houses.

I've dated over 40 people in the past 2 years.
A year ago I tried to eat dinner at a fine dining restaurant but got kicked out because my attire was not formal enough.
Two of my close friends visited me a couple of days ago and we managed to stay at least 6 feet apart from each other while wearing masks to take precaution during this pandemic.
I hate my family. Completely false.
I had dinner with someone I met from Tinder once and I was really looking forward to it because I'd never had a date cook for me before. During dinner I started feeling so sick that I had to leave to go to the bathroom for a while. He tried to food poison me
I was the most fit in middle school.
I once went on a date with a girl who I later found out was a witch.
I used to live in Asia and my dog was eaten for dinner one day. My dog was kidnapped by strangers.
I haven't eaten dinner in 2 years. I've been fasting for the past 2 years and only have an 8 hour window.
My vacation house is burning to the ground as we speak.
I swam with pigs in the ocean.
I was bench pressing one time and I dropped the barbell on my dog on accident because the weight was too heavy. It was very sad but no injuries.

Last year I went on vacation to Hawaii. It wasn't planned but a lot of our friends families also went during the same time as us so we all ended up hanging out together almost every day. I also got to see a turtle.

Table D.1: Stories that achieved more than 60% dominance in our model. Entries have been reproduced verbatim. One entry that only consisted of the letter 'I' is omitted (the subject who was playing the game ran out of time).

Appendix E

Appendix E: Avoided Stories

Avoided Stories
I love going to the ARC. I've used both weight rooms on the right side and also towards the back. There are many amenities and I'm glad that we get them for "free" as a student.
I once went all the way to Toronto to see my favorite artist perform his first ever stadium concert. It was one of the best experiences of my life and I will never forget it.
I was pickpocketed twice when I visited New York.
I went on a trip to Italy and Greece for my 20th birthday!
I eat dinner everyday because I believe that proper nutrition is important especially when working out.
One time I told my friend we should make dinner together, so we went to my apartment to cook and after we spent some time talking he told me he had a crush on me and wanted to ask me out. I had a boyfriend.

I have a friend who loves to eat pasta everyday for dinner. I am the opposite and I can't stand having it often.
My first date was at the movie premier to Toy Story 2.
I have been consistently running/going for walks 3x a week during the past month.
Family is very special. I have not always had the best relationship with them and I think they hold me back sometimes however I know deep down that I will always cherish and miss everyone. Moving on and growing apart is hard. I keep thinking I don't have the perfect family but then again, who really does? Family is what we make of it. Mine is beautiful.
The most stressful exam I had to take was a math proficiency test I took 3 months ago in order waive some Math courses that would've slowed down my degree.
I love my family, but I can't stand them for long periods of time. They annoy me consistently and there is only so much I can handle.
One time I forgot my wallet at home which contained my student ID and my dorm key. I had to ask my roommate to swipe me in for two weeks until I could go back home to obtain my wallet.
Living with my parents and siblings still is very inconvenient. With rules and set times for certain events, such as dinner and game nights, I am unable to have the freedom I would like to have if I were able to live in a dorm.
My friends and I have scheduled a zoom call at 5 pm today, but one of them just canceled because she is out with her boyfriend.

Yesterday I burnt my dinner because I was watching the basketball game for too long. My pizza became a little burnt.
I have only dated 2 people in college.
My first year of housing I lived in a quad. My roommate was horrible and would frequently cause trouble for the entire room. Once such example is when he shattered a glass scale in our bathroom and refused to clean up the mess until several hours later, leaving glass shards all over the bathroom floor.
I lived in a double room in Vista Del Norte last year with my high school friends.
I've never been able to say that "my dog ate my homework." But, I have had my homework eaten by my rabbit. Despite his small size he can do quite a lot of damage with those sharp teeth of his. Putting holes in my papers is a hobby of his.

Table E.1: Stories that achieved less than 15% avoidance in our model. Entries have been reproduced verbatim.