UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

MISINFORMATION DETECTION METHODS USING LARGE LANGUAGE MODELS AND EVALUATION OF APPLICATION PROGRAMMING INTERFACES

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

CATHERINE G. G. DONNER
Norman, Oklahoma
2024

MISINFORMATION DETECTION METHODS USING LARGE LANGUAGE MODELS AND EVALUATION OF APPLICATION PROGRAMMING INTERFACES

A THESIS APPROVED FOR THE
GALLOGLY COLLEGE OF ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. David Ebert, Chair

Dr. Dean Hougen

Dr. Katerina Tsetsura

Dr. Jeong-Nam Kim

Dr. Naveen Kumar

Contents

Abstract

Misinformation has emerged as a pressing public policy concern, prompting transdisciplinary research in the data science field. News journalism provides a foundation for free speech in modern society, yet misinformation in mainstream and independent media through opinionated or biased news can pose dangerous consequences ranging from misunderstanding basic facts to emboldened extremism. Currently, the preeminent tool for misinformation detection is the large language model (LLM) as it is renowned for its ability to capture the context and meaning of textual data. In addition, generative artificial intelligence (AI) models, namely OpenAI's ChatGPT and Google's Pathways Language Model (PaLM), are accessible in an application programming interface (API) form, which can also provide opportunities for automated misinformation detection. Despite advancements in developing effective data science tools for identifying misinformation, there are not many available options, and it is crucial to assess pre-existing tools to determine the most recommended model to pursue for future field research and open-source use in automated misinformation detection. This thesis attempts to evaluate fine-tuned supervised LLMs, AI model frameworks, and unsupervised learning methods to propose an explainable, automated misinformation detection tool that incorporates multiple natural language processing (NLP) dimensions and holistically evaluates trustworthiness in news articles. The study revealed that the Hugging Face LLM RoBERTa with added NLP dimensions as features was the most effective model. Furthermore, it was found that unsupervised learning methods provided valuable insights that eliminated some ambiguity between trustworthy and fake news articles, and AI models

tended to inflate the trustworthiness values of news articles.

Keywords: misinformation, large language models (LLMs), unsupervised learning, application programming interfaces (APIs)

Chapter 1

Introduction

1.1 Background

To begin, the term misinformation is defined as an incorrect or misleading statement that can obscure the truth [29]. A similar infodemic, disinformation, is defined as false information deliberately created or spread in order to cause harm, usually with political, psychological, or social motivations [21]. Misinformation (along with disinformation) is a rapidly growing problem in the modern world today, and while ideally widespread media literacy and regulations on social media content are the most effective methods for helping stop the spread of misinformation [15], there is more demand for taking on massive amounts of misinformation using big data analytical tools through fast and accurate classification of misinformation. Data science, as a STEM field with multiple interdisciplinary applications, can have the technological ability to attain this difficult objective. However, to do this it is important to understand which contributions are the most effective for scientific knowledge and societal purposes, whether these be through the creation of new tools, or the discovery of new data insights, on certain aspects of the misinformation domain. This thesis attempts to quantitatively examine aspects of misinformation that are visible (trustworthiness based on biased news) and invisible (revealing hidden patterns among trustworthy and untrustworthy articles).

This thesis was inspired as a result of issues discovered through the course of an assistantship

project. The objective of this project was to devise a framework for creating trustworthiness scores based on certain dimensions of article texts. It was a requirement that databases of information were to be populated with the scores at an industry-level accelerated pace to meet deadlines, and APIs were used to populate prior fields of information. An API is a type of interface that connects a computer to a server to return information about an input entity. For example, the Google Translate API provides the translation for a given text and target language [55]. APIs are very efficient at providing multi-faceted information about an entity with a quick response time, making them suitable for programming tasks in industry.

Since it was preferable to use an API for populating trustworthiness scores for the database, an option that was considered was the Romanian-based Zetta Cloud TrustServista API, which provided misinformation scores as well as explanations for what dimensions were considered for the scores (i.e., named entities, clickbait, sentiment). However, due to its lack of transparency on cost, this API was not considered and a machine learning approach was eventually used. Besides the TrustServista API, very few APIs were available that could perform the intended task of wholesomely determining a trustworthiness score for any given text, regardless of length. Thus, the lack of availability of efficient automated misinformation detection tools made goals in this project more difficult to accomplish, and the idea behind this thesis was conceived.

1.2 Research Questions

Given the context of the problem presented in the previous section, pressing research questions arise. Additionally, research questions involving the general efficiency of pre-existing tools need to be addressed.

Can a data science tool that rapidly and accurately assesses the misinformation likelihood of news articles, while ensuring transparency through incorporating key NLP dimensions, be developed to assist stakeholders who are potentially

impacted by, or may have a role in, combating misinformation?

Given that there are not many tools available for evaluating the overall trustworthiness of a news source [52], more options should be added to the current competition to have a variety of tools for misinformation detection tasks done on different types of media, especially when it comes to analyzing news media. Creating a data tool that can provide a trustworthiness score for a given news article text, preferably on a scale from 0 to 100 rather than on a binary scale, as well as explain why it gave the text that score could potentially be an effective and explainable misinformation detection tool. To explain the prediction of the trustworthiness score, quantified NLP dimensions will be incorporated and should provide a suitable framework for integrating dimensions of misinformation related to the field of communications, contributing to transdisciplinary work. For the base model of this tool, an LLM will be used as it is currently one of the most effective models for executing NLP tasks; multiple base LLMs will be trained and compared for performance. Misinformation detection is usually a classification-based task, however for the LLMs this problem will be a regression problem as the scale from 0 to 100 provides an opportunity to predict the true likelihood of a news article being misinformation instead of denoting by a label (i.e., 0 or 1) that does not have much middle ground in terms of the real likelihood that a news article is trustworthy. In addition, usage of this tool by the target stakeholders could help reduce the spread of misinformation and its harmful effects on populations, making the tool a potentially valuable contribution to information warfare.

Which dimensions of misinformation will be considered when creating the data science tool for scoring the trustworthiness of a news article?

Besides data science being a field with major contributions to evaluating misinformation potential, misinformation is highly researched in the field of communications as it is largely based on the communicative meaning between words and phrases present as well as the relationship between the author who disseminates the information and the users who consume

that information [10], [11], [12], [43]; these meanings can be capable of being converted into NLP interpretations. 13 dimensions have been proposed for this thesis based on literature review [3], [5], [16], [26], [40], [43], [50], [58], [82] and distribution variability: sentiment, persuasion, exaggeration, context, inclusion of multiple perspectives, use of named entities, use of statistics, referencing of previous articles, term frequency, distraction, verification of claims, logical coherence, and clickbait title. Specific features of these dimensions will be covered in Chapter 2. These dimensions are most indicative of biased or opinionated news sources, which can have a greater likelihood of spreading misinformation and being less trustworthy. These dimensions will also help provide explainability for the proposed data science tool and will be converted into quantitative inputs for the tool using specified vocabulary sets. When it comes to the cumulative quality of these proposed dimensions, these are dimensions that should be universally considered when determining if a given news article is likely to be misinformation as these should not overlap too much in features but provide sufficiently broad accountability in determining trustworthiness.

Can unsupervised learning methods performed on trustworthy and untrustworthy news articles provide any valuable insights for future research on misinformation detection?

Misinformation when presented in an objective tone can pose an invisible but dangerous risk when not held accountable, and the use of unsupervised learning methods has the potential to unveil patterns among this type of misinformation. Although some work may be completed later using a verified true and false statement dataset that can provide contributions to future research on automated fact-checking, this thesis attempts to analyze patterns between the articles that can also contribute to visualizations using misinformation data. These insights can additionally be used in training supervised learning models for improving misinformation detection. Methods including association rule mining, anomaly detection, cosine similarity, t-Distributed Stochastic Neighbor Embedding (t-SNE), and Latent Dirich-

let Allocation (LDA) modeling will be explored in this thesis [9], [32], [39], [53], [73].

Do generative AI models have the potential to detect misinformation accurately?

Relating to the lack of availability of tools like APIs that perform trustworthiness scoring, the options that exist mainly adapt the frameworks of generative AI models, most notably OpenAI's ChatGPT and Google's PaLM, which is the model framework used in Google Bard. AI models such as these are increasingly becoming popular due to their user-friendly nature and ability to inform the user about almost any aspect of existence. However, should they be relied on for misinformation detection tasks, which can have major risks and consequences for society if they are not accurate? These AI models in their API forms will be explored and evaluated in their abilities to rate the trustworthiness of a news article text. These models have the potential to produce similar scores (from 0 to 100) when prompted by the user that can then be compared with the labeled and model scores that were calculated using the NLP dimensions. Responses given by the models appear to be detailed and well-written, including consideration of multiple dimensions to explain why the model gave the news article text the score that it gave. However, despite the seemingly efficient and accurate appearance of the AI models' ability to detect misinformation, it is important to note that using an AI model as an open-source tool to detect misinformation should not be taken for granted. It likewise should be noted that AI models even have the potential to spread misinformation, so there could, ironically, be misinformation in their ability to detect misinformation. Therefore, these models should be evaluated, and this thesis can serve as a preliminary study of how trustworthy these models are for misinformation detection purposes. The AI models' scores and explanations will be evaluated with the proposed model's results using various comparative methods.

1.3 Objectives

There are 3 primary objectives, or components, of this thesis:

1. **Component 1:** Creating and proposing a data science tool that utilizes an LLM as the model foundation and NLP dimensions as supporting explainability and prediction features. This tool will use news article texts as input and will output cumulative trustworthiness scores on a scale from 0 to 100 (0 meaning not trustworthy and high misinformation potential and 100 meaning very trustworthy and low misinformation potential). Multiple base LLMs from the Hugging Face website (`https://huggingface.co/`) including BERT, RoBERTa, and DistilBERT [18], [42], [59], [80] will be trained and compared, and the best performing model that results from this comparison will be used in the tool. A compiled training dataset of news article texts, titles, and origins will be used for this component of the thesis.

2. **Component 2:** Analyzing data patterns between likely trustworthy and likely untrustworthy news articles to discover invisible data patterns using unsupervised learning methods. A separate dataset can be used using true and false statements to eliminate the ambiguity of objective misinformation, however for consistency a subset of the news article dataset will be used for analysis.

3. **Component 3:** Evaluating generative AI models' ability to detect misinformation and score trustworthiness of news article texts and comparing the AI-generated scores and explanations with the true scores. The AI models that will be explored will be ChatGPT-3.5, ChatGPT-4, and PaLM in their API forms. The trustworthiness scores generated from the APIs will be collected and stored in the dataset used for Component 1 since these APIs will be using the same article texts in the training dataset as inputs.

1.4 Significance to Data Science Field

The significance of this thesis to the data science field and general scientific knowledge will help close some important gaps in the research area of automated misinformation detection.

First, Component 1 will help contribute to the creation of a new data science tool to the current competition of available tools used for predicting trustworthiness (or misinformation potential) scores for news article texts. This tool can likewise be open to peer review by experts and professionals in the field who have an interest in creating and evaluating automated misinformation detection tools.

Next, Component 2 will help generate valuable insights from the unsupervised learning methods that can provide significant contributions to the field. The results can have the potential to contribute to future field research on automated fact-checking, improving misinformation detection models, and predicting trustworthiness scoring for news articles. In addition, data visualizations will be made to visualize the patterns detected among the trustworthy and untrustworthy news articles, which can serve as contributions to scientific knowledge about misinformation data visualizations.

Finally, the results from Component 3 should serve as a preliminary study of how reliable or unreliable AI tools can be for misinformation detection tasks, and it is likely that more preliminary studies will come out researching this issue in the coming years.

1.5 Significance to Society and Stakeholders

Besides this thesis' potential impact on scientific knowledge, an impact on society and target stakeholders should also be recognized. Listed below are the target societal stakeholders and how this thesis can benefit them:

- Military agencies (i.e., U.S. Air Force): As the COVID-19 pandemic had a significant impact on supply chains across the U.S., supply chain operations can also impact the military, which has a crucial role in ensuring military readiness and national security interests. Military supply chains can also be impacted by misinformation; for example, if the military were to do business with a company to request products from and there

is misinformation that is spread about the company that inadvertently harms the company's reputation as a potential supplier, the military could refuse to do business with the company based on that false information. Therefore, the military can benefit from this research as it can have the potential to help keep supply chain operations running smoothly by holding misinformation accountable that can sabotage business decisions.

- Government agencies (i.e., DoD, DHS, CIA, NSA): Various government agencies can benefit from this thesis as they too have a role in protecting national security interests and the safety of U.S. citizens. U.S. citizens and government agencies can be subject to damaging misinformation that can affect livelihoods and reputations, so it is important for these agencies to utilize efficient data science tools for automated misinformation detection and to invest in information warfare. A data science tool like the one proposed in Component 1 of this thesis could potentially later be used by the U.S. government to combat and tackle misinformation on a big-data scale.

- Programmers in industry: Industry programmers whose work may involve misinformation detection of texts could be impacted by this research. Industry programmers may work with tools like APIs to populate databases at an accelerated pace to complete deliverable deadlines, and having an API-like tool like the one proposed in Component 1 can be an example of a tool that industry programmers can use for accurate and quick misinformation detection for large databases of texts.

- The public: The tool proposed in Component 1 can also be used for open-source use by U.S. citizens who want to be informed on whether and why news articles that are found on mainstream news sources, independent news sources, or social media are trustworthy or not trustworthy.

Chapter 2

Literature Review

2.1 Research Problem Context

As stated in Chapter 1 of this thesis, widespread social media regulations and media literacy are ideally the most effective methods for mitigating the infodemic of misinformation [15]. A study on the impact of fake news on individuals' partisanship focusing on third-person perspectives revealed the presence of unbiased support for interventions aimed at curbing the spread of misleading information [31].

There are factors that need to be considered when it comes to the news content and the person absorbing that news content. Fake news is less than 1/10th of 1% of all media consumption [79], but despite this figure of fake news consumption appearing quite small, fake news consumption can still be amplified in different contexts. About 1 in 4 Americans have visited a fake news website, and around 6 out of 10 visits to a certain fake news website were from the 10% most conservative Americans [23]. The social media network Facebook has been known to aid in the spread of misinformation [15], [23]. Although the concept of fact-checking appears to be an effective way to discern fake news from real news, on social media networks like Facebook this approach does not reach a wide audience of fake news consumers [23]. Fact-checking has also been shown to not be as effective in mitigating the risk of misinformation spread and information polarization [81]. Additionally, personality factors of people consuming fake news include bias, agreeableness, extraversion, negative

emotionality, open-mindedness, and hours spent consuming news; these factors can lead to media consumers' susceptibility to fake news [6]. Fake news, as it is likewise appealing in attention-grabbing tactics, has also been shown to increase dopamine levels in people's brains during consumption [22], thereby creating an emotional reward for the reader consuming the news content.

It is also important to note which exact definition and form of misinformation needs to be considered in the context of the research problems of the thesis. First, there are specific distinctions between the following infodemic definitions [21]:

- Propaganda: Information shared by the government usually with a political connotation.
- Disinformation: False information that is intentionally disseminated to cause harm.
- Misinformation: False information that is not intentional in harm.
- Malinformation: False information that is deliberately distributed to cause harm to a person's or organization's reputation.

There are yet other types of fake news, including satire, hoax, clickbait, and conspiracy [46], as well as rumors and spam [29]. Opinionated news will be the target category for Component 1 analysis, as this is a form of intentional disinformation and is likely to have more easily recognizable NLP dimensions to embed including high sentiment, high exaggeration, and lack of context that can help predict the trustworthiness scores. For analysis involving a true and false statement dataset, this dataset will contain objective and unintentional misinformation that will be harder to distinguish from factual statements.

The impact of misinformation has shown evidence of being far-reaching and having major negative impacts on society and democracy. Social media usage and fake news absorption have led to increased polarization and misunderstanding in social interactions [4], [70]. Misinformation likewise poses a threat to the institution of journalism as principles of veracity

and objectivity can be violated with the spread of misinformation through the news [81]. One of the most dangerous implications of disinformation, or intentional misinformation, is the use of false pretenses to target people. This can occur in the case of targeting ethnic groups, individuals, or countries through extremist ideologies [20]. Fake news can also kill through pandemics; when inaccurate or downplayed information is being spread in regards to an encroaching pandemic, this can lead to people not taking preventative measures from catching the disease [10]. A final example is the ever-growing problem of well-organized, weaponized disinformation surrounding events of publicity including presidential elections, vaccination, and even infrastructure networks [30]. These harmful campaigns can have the potential to affect public services that citizens are dependent on, and disinformation that can lead to a shutdown of those services can greatly affect supply chain management [2], [30]. Ensuring that supply chain operations, especially for the benefit of the U.S. military, are unaffected by this type of disinformation due to highly efficient misinformation detection has become an important application of this research based on the impacts laid out in this literature review.

Therefore, especially in regards to the double threat of people's susceptibility to fake news and fake news' potential to harm people, it is important to hold misinformation accountable and reduce its spread on a big-data scale, through quick and effective automated means [49]. The longer that the issue of misinformation metastasizes, the more the spread of misinformation becomes out of control, and the less able people can be at distinguishing between fake and real news.

2.2 Applications to Stakeholders

The intended stakeholders of U.S. government and military agencies, industry programmers, and public users in which this thesis work could impact involve many applications. First, when it comes to the overall role of the U.S. military in providing defense for citizens, it

is important that military agencies, especially the Air Force, can be able to meet supply and demand needs to be able to make clear business decisions when building essential military equipment. Thus, misinformation can impact supply chain risk management when false information about a company or country can jeopardize business decisions [2]. Next, when it comes to government agencies, this thesis work could be applied to certain agencies (i.e., NSA, CIA) that focus on defending national security interests against misinformation threats, both foreign and domestic [21], [81].

In the context of programming in industry, not much research has been conducted on misinformation tools helping data scientists in industry, however the accessibility of open-source models and tools such as APIs have been very helpful for programmers in industry to quickly request information about databases of entities on a large-data scale [83]. This thesis work intends to benefit this target stakeholder by providing another easy-to-use tool for misinformation detection purposes. This application is also based on the original background from Chapter 1 that outlined the inspiration for this thesis. Finally, it is important for public users to have knowledge of and access to tools that can verify whether the news that they see on social media, mainstream media, or independent media is trustworthy or not; current open-source fact-checking applications include Politifact, Gossip Cop, B.S. Detector, and Fake News Detector AI [63], [81].

2.3 Misinformation from Communications Perspective

Data science is a field that has the potential to tackle the problem of misinformation from a STEM point of view, however the domain of misinformation can overlap with other academic fields including communications and journalism. Thus, it is important to consider the communication dimensions of misinformation and incorporate them into data modeling through transdisciplinary research.

Numerous dimensions of misinformation exist, both text-based and external. For example,

the use of certain pronouns, politeness, emotion, complex vocabulary, and words indicating uncertainty can mean the difference between fake news and real news [10]. Characterizations of a higher likelihood of misinformation can include a distant speaker-audience relationship, high emotion, a goal to illicit audience action, detailed information, and a high number of participants and sources cited [11]. Factors including diffusion scope (how broad the audience is), speed (number of users reacting within a certain timeframe), and shape (broadcast or human-human transmission) have been used to determine misinformation through analyzing clusters of users reacting to certain tweets on the social media network X, formerly known as Twitter [12]. Sentiment is possibly the most important textual dimension of misinformation, and the use of emotion in misinformation can lead viewers to be more engaged and susceptible to misleading messages [43].

So how can communication dimensions be incorporated into a data science model? This is through the textual analysis technique of NLP, which is where a computer is trained to interpret human language; this is usually done through converting words into numerical representations, also known as word embedding [76]. NLP tasks such as stance detection, rumor detection, and sentiment analysis can be performed by converting certain word vocabularies into numeric inputs [66].

Regarding misinformation detection APIs that are restricted in availability, these can have the ability to not only provide a trustworthiness score, but also take into account which communication dimensions were taken into consideration when reading an article text. The Romanian-based TrustServista API has this capability and can take into account factors such as named entities, context (who/what/where), clickbait title, and sentiment, however it is acknowledged that there has not been much academic research conducted evaluating this API due to lack of transparency [69].

2.3.1 Concepts Behind Natural Language Processing Dimensions

Considering which communication dimensions, to be translated into NLP dimensions, are most important for identifying what can be biased or opinionated news, a total of 13 dimensions stand out that can be identifiable using specified word embedding vocabularies. While not all of these dimensions are present in every news article, these dimensions should be universally considered when evaluating the news article for misinformation detection based on this literature review.

1. **Sentiment:** Sentiment is potentially the most characteristic dimension of subjective news [43], and it emotionally impacts readers to believe in false messages in the article.

2. **Persuasion:** This dimension is the same as bias and opinion; language that attempts to get the reader to align with the article's point of view is a very likely indicator of misinformation [26].

3. **Exaggeration:** Exaggerated and/or outraged language is also indicative of misinformation [5].

4. **Context:** The more context that there is with cited persons of interest, the less likely that there is misinformation present in the article [3].

5. **Inclusion of multiple perspectives:** Multiple perspectives that take in different points of view from people of interest add more layers of context to the article and reduce bias [3].

6. **Named entities:** Named entities can be very important for adding more context to a news article and decreasing the likelihood of misinformation, especially if there is a greater diversity of named entities [16]. The less diversity in named entities present in an article, the more likely it is for misinformation to be present in the article.

7. **Use of statistics:** Statistics can be very effective tools for clarifying facts in a scientific manner and can add more context to a news article's message. Even mentions of dates and monetary figures can fall under the umbrella of statistics as these can clarify exact timeframes and figures. More statistics means more likelihood that a news article is trustworthy. This dimension is not commonly thought of when it comes to disinformation detection, however it is important enough to provide well-rounded context to an article [3].

8. **Referencing of previous articles:** As an alternative to cross-referencing, if the article references certain information from a previous news article or interview, or evidence from a previous report, that can add to the reliability of a news article [82].

9. **Term frequency:** If there are terms that are irregularly compared to words that are normally used, this can lead to more likelihood of misinformation [58].

10. **Distraction:** Words that indicate an attempt to distract or deflect from the main topic of the article can lead to a greater likelihood of misinformation [40].

11. **Verification of claims:** As an alternative to fact-checking, this dimension can provide factual clarity to claims that are made in a news article and add to the trustworthiness of the article [82].

12. **Logical coherence:** Words such as "first", "then", "therefore", and "consequentially" indicate a logical flow of ideas and subjects in a news article, which can add more context [82].

13. **Clickbait title:** As it is the only dimension related to the article title rather than the article text, words present in the title that indicate clickbait or exaggeration can be indicative of an article not being trustworthy [50].

2.4 Related Work

This section will cover a generalized overview of research in the data science field on misinformation detection. More related work regarding the specific components of the thesis will be covered later in this chapter.

With the rise of misinformation as an issue to be tackled, it has also risen as an issue of popularity within the scientific community, especially in the field of data science. In previous field research, it has generally been established that supervised learning is required to train a model to predict misinformation, and simple machine learning models including Naive-Bayes classifier, support vector machine, random forest, and XGBoost have been used and compared in their accuracies for classifying misinformation [57]. As the transformer model is the most advanced model today that can predict misinformation [74], before its introduction deep learning models such as neural networks were the most efficient machine learning models available for this task. For example, a Long Short Term Memory Recurrent Neural Network (LSTM-RNN) was used for classifying misinformation on a binary scale (i.e., 0 for misinformation, 1 for not misinformation) [14]. Deep learning discriminative, generative, and hybrid models including recurrent neural networks (RNNs), deep belief networks, and convolutional restricted Boltzmann machines have been used and surveyed for misinformation detection [29]. Text pre-processing mechanisms including regular expressions, lemmatization, stop word removal, conversion to N-gram vectors and term frequency-inverse document frequency (TF-IDF) sequence vectors have been used for converting text into vectorized inputs for deep learning models imported from the Python Keras package for fake news detection [38].

Not only various models have been used for misinformation detection, but also different types of data have been utilized in previous field research. From politics to health, social media post format to news article format, many categories of data have been used for mis-

information detection. Machine learning algorithms have been used to filter out opinion spam, or comments that are repeatedly posted to push an opinion or viewpoint [49]. The topic of COVID-19 has become a recent catalyst for misinformation research, for example on YouTube videos as data containing COVID-19 misinformation [60]. Classification data has even included COVID-19 misinformation related to cannabidiol products and warning letters from the Centers for Disease Control and Prevention advising against using cannabis products to cure COVID-19 [71].

Research has also been conducted on misinformation detection data that is not in the English language. Misinformation is not only a problem in the U.S., but it is also a problem with a global impact. Misinformation that is not in English can be harder to detect due to the presence of the foreign-language effect obscuring intuition when it comes to determining misinformation potential [72].

2.5 Large Language Models

2.5.1 Theory Behind Supervised Learning and Transformer Model

Supervised learning is a machine learning approach that is designed to train a model to correctly predict output values given specified input values. This is usually by training, or making the model learn how to predict correctly, by using a pre-labeled training dataset [57]. Supervised learning models have existed for decades for machine learning purposes, including logistic regression, Naive Bayes, support vector machines, random forests, and decision trees, and these can be used for classification and regression tasks [8]. Specifically, detecting misinformation has been commonly used as a supervised learning task, with an example being training a model to classify whether a news article is true or fake [57].

The large language model (also known as the transformer model), which is a more advanced supervised learning model, was only recently invented in 2017 by Google [74]. Figure 2.1

shows the general architecture of the transformer model. The transformer architecture is divided into an encoder and a decoder. The encoder consists of self-head attention layers and feed forward network layers that accept input sequences. Using these transformed inputs from the encoder, multi-head attention layers (and additional feed forward network layers) in the decoder help capture the relationships among the positional embeddings in the inputs, and then generate the outputs.

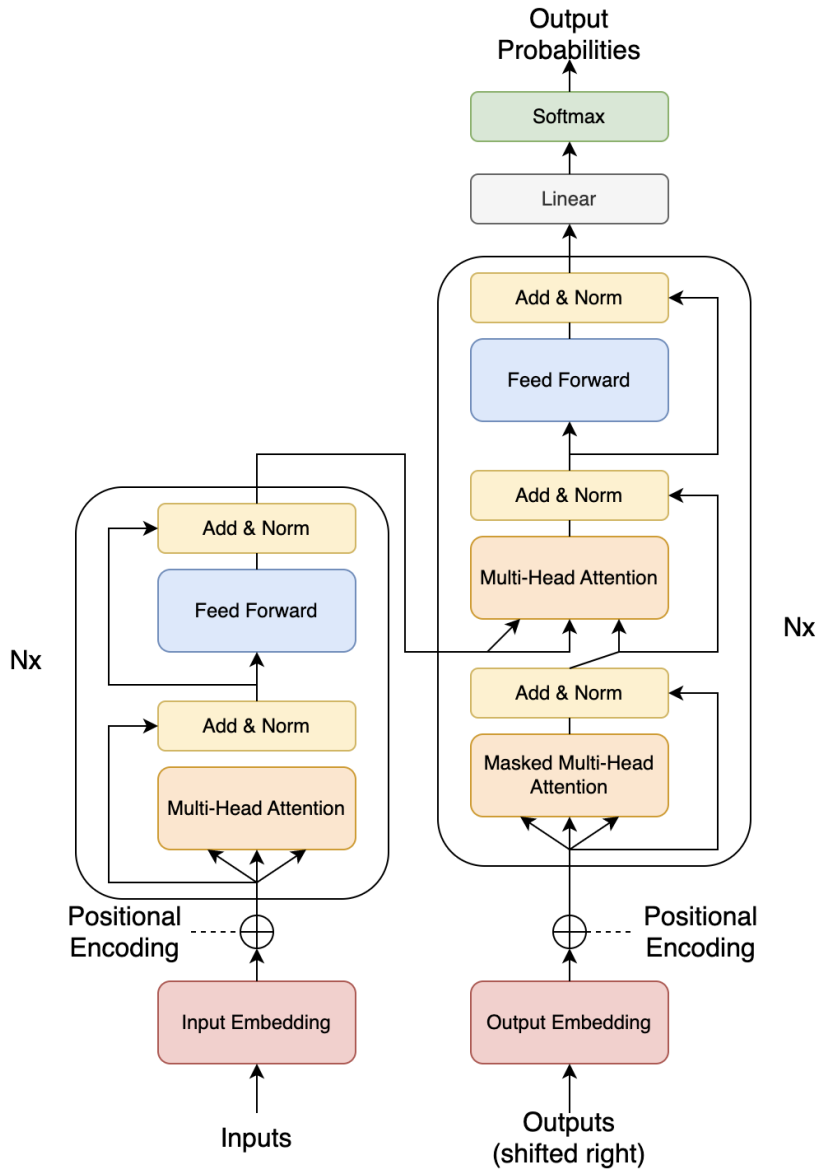


Figure 2.1: Transformer Model [74]

It is very crucial to note that transformer models in their inherent training nature are not necessarily supervised. The original transformer model developed by Vaswani et al. (2017) was trained in a semi-supervised setting, meaning that it was trained on a small amount of labeled data but used the rest of the data that was unlabeled to make predictions. Similar methodologies were made for training the LLMs that will be discussed in this section. However, in the context of this thesis, the LLMs will be regarded as supervised learning models; although the models are pre-trained in a non-supervised setting, fine-tuning the LLMs to perform NLP downstream tasks will require labeled data and a supervised learning approach [25], [42].

2.5.2 Bidirectional Encoder Representations from Transformers

Bidirectional Encoder Representations from Transformers, also known as BERT, was first proposed by Google in 2018 [18]. This model was pre-trained using unlabeled data in an unsupervised setting to perform NLP tasks including masked LM and next sentence prediction [18]. It also incorporated significant improvements upon previous language representation models, which had flaws of unidirectional NLP tasks and left-to-right context evaluation [18]. BERT's ability to fuse the left and right contexts of sentences together to evaluate text holistically, and its versatility in its ability to do text prediction, question answering, and text classification [18] made it a powerful tool for various NLP tasks.

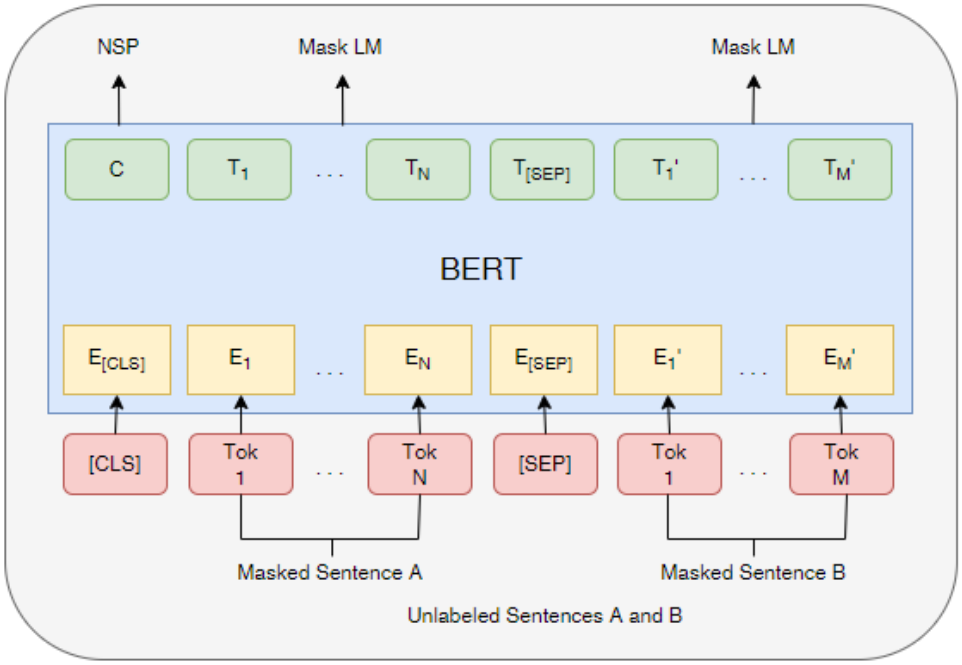


Figure 2.2: BERT Model Architecture (Pre-Trained) [18]

Figure 2.2 shows the architecture of the pre-trained BERT model. Since BERT is trained to do next sentence prediction (NSP) and masked language modeling (MLM), these involve randomly masking parts of the input sequences to train the models to guess the next sentence or hidden tokens. Using the original transformer architecture from Vaswani et al. (2017), the input embeddings that are to be put into the BERT model are the sum of the token embeddings, the segmentation embeddings, and the position embeddings [18].

2.5.3 Robustly Optimized BERT Approach

Robustly Optimized BERT Approach, also known as RoBERTa, was proposed by Facebook AI Research in 2019 as an improvement upon the original BERT model [42]. The model was pre-trained using unlabeled data but fine-tuned using labeled data [42], which is also known as self-supervised learning. Liu et al. (2019) argued that BERT had certain performance training flaws that could be improved; the upgrades that RoBERTa incorporated included training the model longer, using larger batches and sequences of textual data, and dynam-

ically changing the masking pattern applied to the training data [42]. Incorporating these changes showed that the accuracy and performance of NLP downstream tasks on various open-source datasets improved significantly using RoBERTa compared to BERT; the longer the model was trained, the more these results reflected that performance [42].

2.5.4 Distilled BERT

A distilled version of the original BERT model, known as DistilBERT, was proposed by Sanh et al. (2019). Since BERT was trained on a larger number of parameters and required a large amount of data to be trained on, it was computationally expensive to train and perform NLP tasks, therefore DistilBERT was introduced as a computationally faster and lightweight alternative [59]. DistilBERT is 40% smaller than BERT and retains 97% of BERT's performance; its accuracy in performing downstream NLP tasks is similar to BERT [59]. As seen in Figure 2.3, the architecture functions by incorporating less transformer layers, containing the same multi-head attention and feed forward mechanisms, than BERT-base's original layers while preserving BERT-base's efficacy.

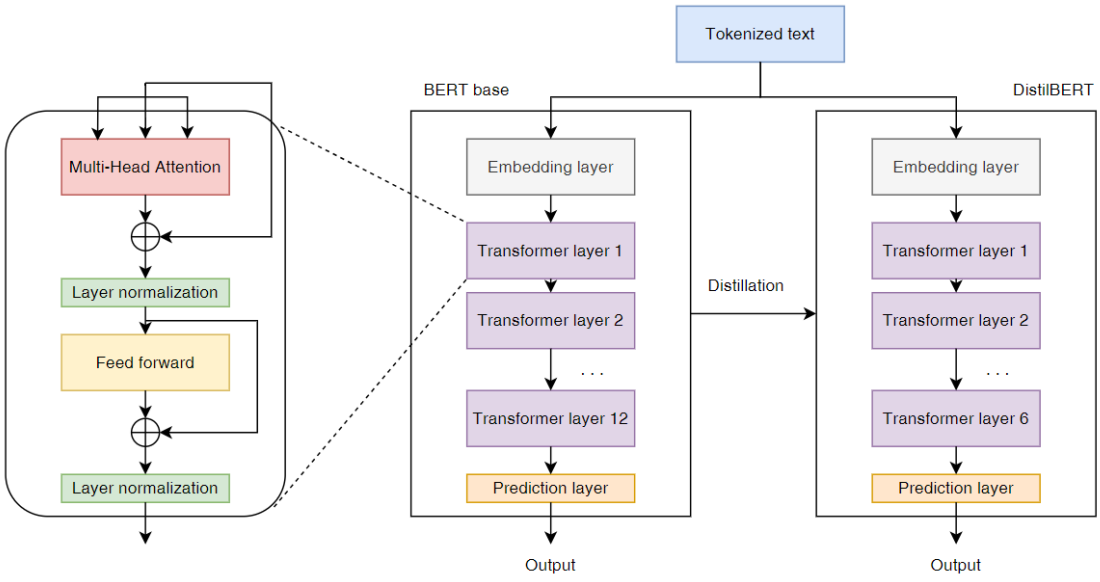


Figure 2.3: DistilBERT Model Architecture [1]

2.5.5 Applications of Large Language Models to Natural Language Processing Tasks

BERT, RoBERTa, DistilBERT, and many other LLMs are available for open-source use on the Hugging Face website `https://huggingface.co` [80]. These models can be downloaded using the Transformers package in Python and fine-tuned to perform any NLP task including chat generation, sentence prediction, text classification, and text summarization. Fine-tuned models designed to perform specific NLP tasks can also be deployed to the website. Generally, the maximum number of input tokens that these models can process is 512 [18], which is equivalent to around 400 words. OpenAI's ChatGPT, although regarded as AI, is at its foundation an LLM as it has the ability of being a chatbot with multiple NLP-based applications including assisting with writing papers, assisting with programming code, and being an interactive learning tool by providing answers on any topic [45]. In addition, LLMs have been used for misinformation classification tasks on textual data, and these transformer models have overall provided good results for performing these tasks [13], [33], [54]. Finally, comparing multiple LLMs including BERT, RoBERTa, and DistilBERT to see which model performs most effectively in NLP tasks such as misinformation classification has been utilized, as model performance comparison is a common practice in the data science field [42], [59], [57].

2.5.6 Applications of Large Language Models to Application Programming Interfaces

APIs have likewise made an impact on AI accessibility [51]. ChatGPT currently has a pay-as-you-go API with different models depending on the intended maximum number of tokens used. As ChatGPT is also a large language model, the ChatGPT API serves as an example integration of LLMs and APIs to not only provide multi-dimensional information from the API's website server, but also actual model predictions from the LLM itself.

2.6 Unsupervised Learning

2.6.1 Theory Behind Unsupervised Learning

Unsupervised learning is a different type of approach in machine learning that sharply contrasts with the approach of supervised learning. In supervised learning, the model is trained based on pre-labeled data, however in unsupervised learning the model utilizes unlabeled data, allowing the model to generate patterns and insights among the data. Common unsupervised learning tasks include clustering of data points and multi-dimensional feature selection [35].

2.6.2 Association Rule Mining

Association rule mining is generally performed on transactional data to determine which items in a transaction go together, and this method can be found in open-source Python packages including Apriori or FP-Growth [39]. Two common examples of association rule mining are determining which items in a grocery shopping list are generally bought together and determining which movie or TV show choices on a streaming service are watched together. Data preprocessing for this method is usually done by generating frequent itemsets among the data through one-hot encoding the terms that are present or not present in the transaction (i.e., assign a value of 0 if not present and assign a value of 1 if present). Then given these frequent itemsets, association rules are generated based on how commonly the items in the itemsets appear and which items are most closely associated with one another in the transactional data. Figure 2.4 shows an example of general processing with association rule mining.[1]

[1]Reference link for the image in Figure 2.4: `http://www.big-data.tips/association-rules`

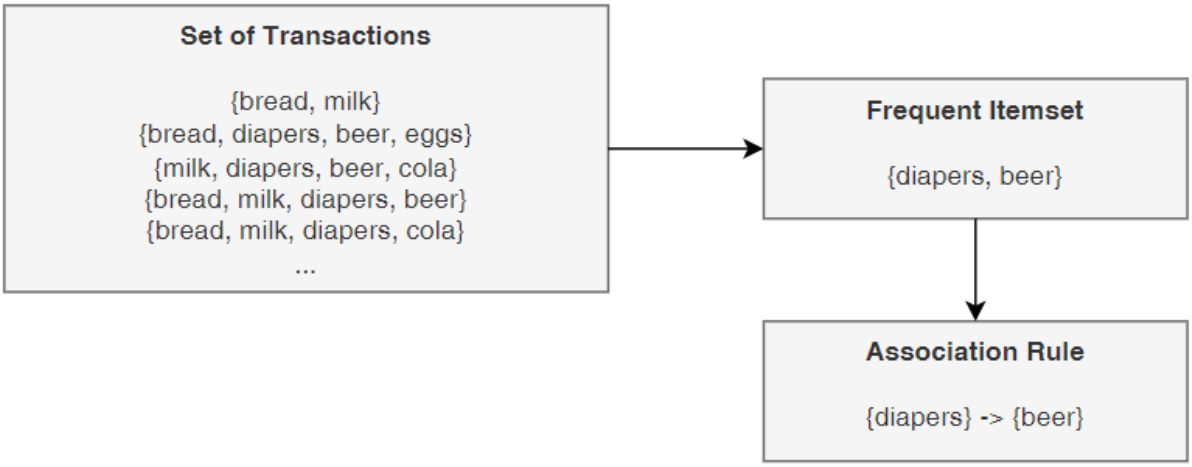


Figure 2.4: Association Rule Mining Example

2.6.3 Anomaly Detection

Anomaly detection is the unsupervised learning technique of finding patterns in data that do not conform to normal behavior [9]. Data preprocessing is done by converting data into a vectorized matrix using TF-IDF, bag-of-words, or other word embedding techniques. After defining the number of reduced features for analysis and the percentage of intended outliers, or anomalies, in the data, an isolation forest model is then initialized and trained to determine which data points are considered anomalies in the data; an isolation forest is an ensemble of isolation trees that are built for a dataset, and the anomalies that are determined by the isolation forest are the observations that have the shortest average path length on the isolation trees [41]. For example, setting the model with a contamination of 0.05 means that 5% of the data points will be designated as anomalies based on their reduced feature values, and data visualizations illustrating anomaly detection can show some data points that are colored differently compared to the majority of the data points.

2.6.4 Cosine Similarity

Cosine similarity is a text similarity metric that is commonly used for text analysis. This metric models a text document as a vector of terms, and the similarity between any two documents can be derived by calculating the cosine value between the two documents' term vectors. Cosine similarity can have applications to semantic similarity, or how similar texts are regarding contextual meaning [53]. The formula below represents the cosine similarity between two text vectors $\mathbf{A}$ and $\mathbf{B}$, where $\cos(\theta)$ is the cosine of the angle between the two text vectors, and $\|\mathbf{A}\| \cdot \|\mathbf{B}\|$ are the Euclidean norms of the vectors $\mathbf{A}$ and $\mathbf{B}$.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \cdot \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i \cdot B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}}$$

2.6.5 t-Distributed Stochastic Neighbor Embedding

Stochastic neighbor embedding is a dimensionality reduction approach that is used to calculate the Euclidean distances between data points as reduced feature similarities [73]. t-SNE is an expansion of that, as it provides a solution to visualizing high-dimensional data in a 2- or 3-dimensional space, resulting in the generation of differentiated clusters of data points based on their reduced feature values [73]. An example of such visualization, data visualizations showing results of t-SNE analysis can display differently colored or labeled clusters of data points in a 2- or 3- dimensional scatter plot [73].

2.6.6 Latent Dirichlet Allocation Modeling

LDA modeling, also known as topic modeling, is a common unsupervised method used for generating topics based on an input corpus of text documents [32]. The LDA model can be found for open-source use in the Gensim Python package. The LDA model assumes that the corpus contains texts over a variety of topics and that each topic contains a certain distribution of words. The model iterates through each of the documents until a certain

number of topics with keywords are generated, and subsequently the topics can be inferred based on the keywords found by the model.

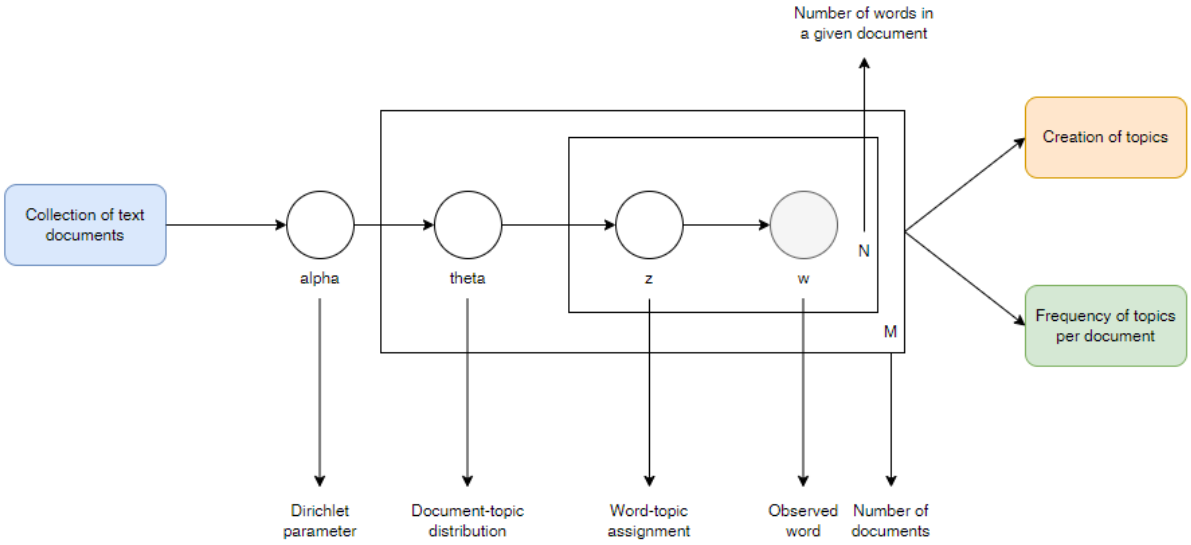


Figure 2.5: LDA Model [61]

2.6.7 Applications of Unsupervised Learning to Misinformation Detection

Some of the aforementioned and other unsupervised learning methods have been used in misinformation detection [27], however research using these models has not been as extensive compared to research using supervised learning models. Association rule mining has been used for determining patterns among misinformation in women's health tweets [37]. Various anomaly detection models have been used for classifying financial misinformation [64]. t-SNE clustering was used to plot the topic clusters of texts to show visual relationship patterns between topics in misinformation data [14]. LDA modeling in conjunction with calculating cosine similarity scores has been implemented in misinformation data training and has been shown to produce accurate results [46]. Topic modeling has also been shown to produce topics that are likely to be associated with misinformation claims [81]. Given these studies, using these unsupervised learning methods to determine hidden data patterns between true

and fake news articles could potentially be of great value.

2.7 Generative Artificial Intelligence Models

2.7.1 Theory Behind Artificial Intelligence

AI is the concept of intelligent machines or computers designed to emulate human intelligence [36]. As AI is based on neuroscience and how the human brain functions, neurons were able to be mathematically implemented as early as 1943 [44]. This breakthrough led to the eventual development of the artificial neural network (ANN) in the 1980s, which is currently the most widely used model in AI research [19]. Prior to the rise of AI applications like ChatGPT in late 2022, advancements in supercomputing power and big data technologies led to the acceleration of AI development, thereby making effective cognitive computing easier [19]. Deep (machine) learning algorithms like ANNs can also be credited for the renewed success of AI research endeavors before the popular emergence of ChatGPT [19].

Machine learning and neural networks reflect a cognitive science application of AI, while NLP is part of a natural language interface application of AI, therefore these subjects are not the same as, but are rather subareas of, the general field of AI [36]. Over the years, AI has evolved to be implemented through LLMs and NLP [56]. This has led to the emergence of generative AI, a class of AI language models that can create new data based on patterns and structures learned from existing textual data; ChatGPT is an example of generative AI [56]. The transformer model architecture [74] has likewise been a staple foundation of AI models; this type of model helped introduce solutions to limitations of previous sequence-to-sequence models like the convolutional neural network (CNN) and the recurrent neural network (RNN) [56]. The recent advancements in generative AI, including having a language model as its base, have led to AI gaining much popularity through its superior performance as a generative chatbot and easy accessibility [56].

2.7.2 OpenAI ChatGPT

OpenAI's ChatGPT was released to the public in November 2022 as a user interface chatbot, using its version 3.5 as the model architecture [56]. ChatGPT was trained on a large corpus of text data and fine-tuned on a specific task of generating conversational responses, which allows it to generate human-like responses to user queries [56]. ChatGPT also has an API where the user can specify the version and number of input tokens desired for data collection. ChatGPT is known to have some limitations including being trained on data that is only as recent as 2021 and being trained on data that can contain biased language [17], [56]. Other significant concerns about using ChatGPT include data privacy, production of factually inaccurate or unreliable responses, and inability to capture the contextual understanding of prompts [56].

2.7.3 Google PaLM

Google's Bard was released in March 2023 and is a user interface chatbot similar to ChatGPT. Bard differs from ChatGPT such that the data that Bard is linked to is from Google's expansive engine search network, while training data for ChatGPT was not necessarily linked to the internet [77]. The language model foundation for Google Bard is the Pathways Language Model, also known as PaLM [77]. While Bard is Google's user interface version of its AI, PaLM 2 (released May 2023) is Google's API version of Bard.

2.7.4 Applications of Using AI Models for Misinformation Detection

GPT-3, a previous version of the current GPT model, has been observed as being a potential spreader of misinformation [67]. Some AI models and social media networks have APIs for easy data collection, however these can have the potential to spread disinformation [48]. In a preliminary study, 72% of verified claims were correctly identified by ChatGPT [28]. This

statistic is modestly good, however it is possible that this number is not higher due to some of the claims originating after the year 2021, where ChatGPT is most recently trained on data pertaining to that period. A study involving evaluating ChatGPT-3.5, ChatGPT-4, Bard, and Microsoft's Bing AI showed that these had a moderately promising ability to classify misinformation accurately, with ChatGPT-4 having the best accuracy with 71% and the average overall accuracy among the AI models being 65% [7].

2.8 Research Gaps

After evaluating the literature review that has been conducted relating to this thesis, there are some research gaps that this thesis can address. First, since most misinformation classification tasks using LLMs have used binary labels [14], utilizing a regression-based framework instead would allow the base model of the proposed tool to assign scores on a comprehensive scale from 0 to 100, using the NLP dimensions of trustworthiness for more sophisticated analysis. The NLP dimensions can also help address a gap regarding the Hugging Face LLMs, which can only input a maximum of around 400 words. Thus, holistic evaluation of a news article being greater than 400 words is difficult using only an LLM; there must be other features, taking into account the entire context of the article text, to help improve the predictability of the trustworthiness scores. Explainability in misinformation classification tools is another important research gap to address. AI models can have the power to spread and counter misinformation, but it is important to promote greater transparency and "human-in-the-loop" elements in future research [34]. Including explainability in the proposed tool to justify which NLP dimensions were present or not present in an input text would make a transparent and trustworthy addition, as some LLM-based AI systems currently lack explainability; this can lead to users not being able to understand why the model is making the decisions that it is making and consequently, trust in the technology can be diminished [56]. A research gap that was also discovered was that regardless of the development of

pre-existing tools used for misinformation detection, the data that these tools are trained on will have to be periodically updated to include more up-to-date information [24], [63].

Some research has been done with regard to misinformation characteristics through association rule mining, cosine similarity, t-SNE, and anomaly detection, but this research with such unsupervised learning methods is not as extensive as research using supervised learning methods. Although a separate dataset of true and false statements could be used for the unsupervised learning approaches will likely be limited to one topic with a sufficient amount of data, these unsupervised learning approaches can be used on other pre-labeled topic datasets, including FEVER and LIAR [68], [78]. However, the data that will include some news articles will likewise be valuable to determine which invisible characteristics could distinguish which news is true and which is fake. Such characteristics could include a reduced dimension value, a topic keyword, a distribution having a center significantly different from another distribution's center, a cosine similarity value, or a cluster of data points that are significantly different from another cluster meaning a greater or lesser likelihood of misinformation for a given news article. Data visualizations will likewise be instrumental in illustrating these revelations in distinguishing the true and fake news. Some research on such data visualizations (not pertaining to misleading data visualizations or images in news) has been done [65], [82], however there is still a gap when it comes to visualizing vectorized or textual differences between real and fake news.

Finally, ChatGPT and PaLM are recent AI models that currently appear to have the potential to detect misinformation if prompted with an input text. Despite some research having been done on their ability to detect misinformation [28], [7], more studies need to be conducted to determine whether these AI tools can be trusted for misinformation detection tasks and which advantages and disadvantages these tools have when detecting misinformation. These aspects can pertain to the classification accuracy as well as the perceived explainability of these AI models.

2.9 Summary

The discussion of this literature review helped integrate communication dimensions of misinformation with the field of data science to open the possibility of creating data tools that can identify misinformation potential. There are also the explorations of unsupervised learning and evaluating AI models that have had some but not sufficient research done on their capability to effectively identify misinformation. This thesis will help address significant research gaps connected to these areas in the domain of automated misinformation detection.

Area	Related Work	Research Gaps	Potential Solutions
LLMs	-BERT, RoBERTa, and DistilBERT used for misinformation classification and NLP tasks -Binary misinformation classification tasks common	-More tools needed for automated misinformation detection tool -Lack of explainability in tools for misinformation detection -Regression tasks not common -Hugging Face LLMs take a maximum of 512 input tokens, which may not be sufficient for long news article texts	-Create regression-based tool for trustworthiness scoring of news article texts -Evaluate texts holistically and transparently using NLP dimensions
Unsupervised Learning	-Some work done on misinformation domain using association rule mining, t-SNE, etc. -Some work done on data visualizations differentiating characteristics of real vs. fake news	-Misinformation detection research using unsupervised learning not as common -No work done on data visualizations using unsupervised learning methods	-Conduct simple unsupervised learning approaches on news article texts -Create data visualizations to illustrate insights from unsupervised methods
AI Models	-Some studies show that ChatGPT and PaLM are moderately accurate at classifying misinformation	-Not sufficient evidence to show which, if any, AI models can be trusted for misinformation detection tasks due to their recent innovation -Lack of explainability	-Collect AI trustworthiness scores and compare them with true scores

Figure 2.6: Tabular Summary of Literature Review

Chapter 3

Supervised Learning Using Large Language Models and Natural Language Processing Dimensions

3.1 Overview

The first type of model that will be analyzed in this thesis is the fine-tuned supervised learning LLM. Pre-trained LLMs are usually trained in a self-supervised manner, however fine-tuned LLMs can be trained in a supervised manner to perform downstream NLP tasks. Supervised learning models are trained to make predictions, and LLMs are complex, versatile NLP models that can execute analysis using the forward and backward context of input texts. LLMs have been shown to produce promising misinformation detection results, however when compared to the approaches of unsupervised learning and AI, is supervised learning using LLMs necessarily the most effective approach? Or do other approaches reveal features that can predict misinformation more accurately? That is what this thesis will attempt to resolve.

Thousands of news article texts covering a variety of topics will be used for training the LLM, as it is important that the LLM be trained on as large and diverse a dataset as possible. The Hugging Face LLMs BERT, RoBERTa, and DistilBERT will be trained on the dataset and compared to determine which model has the better performance in predicting trustworthiness scores. The models will also be trained regression-wise to produce trustworthiness scores

between 0 and 100 (0 meaning low trustworthiness and high misinformation potential and 100 meaning high trustworthiness and low misinformation potential) based on communication dimensions of misinformation that are easy to embed as NLP dimensions; these dimensions are easier to identify as features of biased or opinionated news. These NLP dimensions will serve as quantified features that will help make the trustworthiness score predictions more accurate.

LLMs usually take a maximum input of 512 tokens, and a significant portion of these news articles are longer than 512 tokens, thus the model has to truncate the input texts and not incorporate the entire context of the articles. By themselves, these truncated texts will not help the LLMs predict the trustworthiness scores accurately, therefore the NLP dimensions will be needed for capturing the entire context of the articles and fed into the LLMs. Therefore, it is important that the chosen base model for the data science tool be able to evaluate entire news articles, regardless of length, to bypass the limitations of the LLMs to make accurate misinformation potential predictions.

3.2 Natural Language Processing Dimensions

13 NLP dimensions were chosen for creating the trustworthiness scores. The minimum cumulative trustworthiness score that can be given by the model is 0, and the maximum cumulative trustworthiness score that can be given is 100, as the separate dimension scores will be summed together. Below is why these dimensions were chosen, which textual features are indicative of these dimensions, and how much they will be weighted. All dimensions except for referencing of previous articles, distraction, verification of claims, and clickbait title were weighted a 10 as these dimensions appeared to have the most relevancy and impact on trustworthiness, which will be discussed further in Section 3.5.4.

1. **Sentiment:** Are there words that indicate praise, fear, anger, etc. for the person or topic of interest? Sentiment is an important indicator of biased or opinionated news,

and due to its impact on whether an article is trustworthy or not, it will be given a maximum weight of 10.

2. **Persuasion:** Are there words indicating biased rhetoric or attempting to coax the reader to accept the article's point of view? Also known as a dimension that is indicative of bias or opinion, language that attempts to get the reader to align with the article's point of view is a very likely indicator of misinformation. This dimension will be given a maximum weight of 10.

3. **Exaggeration:** Are there words or punctuation that indicate exaggeration or outrage? Exaggerated and/or offensive language is a very definitive indicator of misinformation and untrustworthiness of a news article, and its impact means that it will be assigned a maximum weight of 10.

4. **Context:** If citing potential misinformation, is the person of interest saying it or is the article saying it? Quotation marks are indicators that the article is directly citing a person of interest, and context that includes whether the article is saying a statement or if the person of interest is saying a statement is important in determining whether the article is providing a platform for misinformation in its viewpoint. If the article provides a biased viewpoint and does not objectively cite a person's misinformed statement, then that can be indicative of misinformation. This dimension will be assigned a maximum weight of 10.

5. **Inclusion of multiple perspectives:** Are there words that indicate that someone said, suggested, explained, etc. in the article? Multiple people being cited in the article that have different viewpoints and defenses make the article more objective, less biased, and with more context. This dimension will have a maximum weight of 10.

6. **Named entities:** Are there multiple named entities? A wide diversity and usage of named entities, including people, organizations, dates, and locations are important for

providing additional context to the article. This dimension has a large impact and will be given a maximum weight of 10.

7. **Use of statistics:** Are there numbers or figures that are present in the article? Usage of statistics provides even more context to a news article, but more so from a numerical or mathematical standpoint. Polls, surveys, data trends, monetary figures, and other numerical points can contribute a well-rounded, factual context, and can have a great impact on determining whether a news article is trustworthy or not. This dimension will be given a maximum weight of 10.

8. **Referencing of previous articles:** Are there words that indicate previous stories, editorials, or op-eds? Some articles can cite previous reports, op-eds, and articles that can give some context to the subject matter of the news article. However, since this dimension may not be present in all news articles, this dimension will not have much impact and will be assigned a maximum weight of 2.

9. **Term frequency:** Are there unique terms or words that compare to the rest of the corpus of news articles? Unique terms that may be different from normal words used in a news corpus can be indicative of misinformation. These uncommon terms may raise concern about how biased or opinionated the news article is. This dimension will have a maximum weight of 10.

10. **Distraction:** Are there words that indicate distraction or contradiction from the main topic of the article? Some articles may attempt to distract the reader by using certain words that divert from a given topic of interest that may be important to address. However, this tactic is not used in all articles, therefore this will have a maximum weight of 2.

11. **Verification of claims:** Are there words that acknowledge that a claim has been verified, alleged, debunked, etc.? While fact-checking objective statements is difficult,

there can be ways to quantify words that acknowledge that certain claims were alleged, confirmed, debunked, etc. such that it can add an alternative. However, since some articles do not have this, it will be given a maximum weight of 4.

12. **Logical coherence:** Are there words that indicate logical sense or flow? A logical flow of the article or viewpoint can provide logical context and can also indicate whether the text is well-written. This dimension will have a maximum weight of 10.

13. **Clickbait title:** Are there words in the article title that indicate potential clickbait? A clickbait title can be indicative of an article that may likely not be trustworthy. This dimension will have a maximum weight of 2.

Score: 35
Article Title: FULL TEXT: MILO On The Supreme Court And Why Conservatives Must Vote For Trump – Breitbart

Welcome to the Dangerous [redacted] Tour! I am JUDGE MILO, the most fabulous supreme justice on the Internet. Friends and foes simply call me MILO — and that's not because they can't pronounce my last name. It's because I'm famous. Liberals are afraid to say many hard last names, because if they say it wrong it's racist, and if they say it correctly it's cultural appropriation. But I'm a European, so they don't have any of those problems with me. Today is Yom Kippur, the Jewish day of atonement. Jews everywhere are apologizing for their sins. I'm blessed to only be because God doesn't have time to listen to my full list. I was raised as a Catholic, of course. To be honest I only mentioned Yom Kippur to make the [redacted] on Stormfront froth at the mouth. They love to hate me! I'm convinced Stormfront is a creation of the CIA, you know. It's too hilarious to be written by real white supremacists. I'm having a great tour of the south, although I'm starting to wonder if country music is a Clinton plot to have me commit suicide. Sorry, that [redacted] just doesn't travel outside the US. I'm having huge turnouts and massive viewership of my talks on YouTube. I think some of the viewers watch to see if I will crash and burn. It's the reason southerners watch NASCAR but also Hillary Clinton speeches. But I'm sorry to disappoint, I'm having a ton of fun on tour and everything is going great. I'm staying healthy, safe and happy. I even have my own security detail. Unlike Hillary's, none of them are seizure doctors in disguise. Anyway. Let's grab the subject at hand right by the [redacted]. We are officially less than thirty days out from the presidential election. We've already seen over the weekend how the Clinton Campaign along with their unofficial campaign wing, the Republican Party, have tried to smear daddy. The tape they released didn't affect the polls in any meaningful way. The only people shocked that Donald Trump talks like a blue collar guy were apparently his fellow Republicans, who rushed to cover the ears of their wife's boyfriend's children. A lot of you are on edge. You're asking yourself, "what's next?" What will the next tape hold? And what kind of damage will it do? We have to be realistic: there probably are more tapes of Daddy. He worked in TV for many years, and let's face it, Daddy likes to banter! For some Trump supporters, especially young people who haven't experienced many dirty elections, this is a shaky time. Our candidate has an unholy coalition against him. The Clintons and their smear machine, the mainstream media, Silicon Valley, and the Republican establishment are all out to stop Trump. We now know from Wikileaks that The New York Times gave Bill Clinton interview questions in advance.

Figure 3.1: Example of Article Text with Low Trustworthiness Score

Figure 3.1 shows an example of an article text from the training dataset that had a low trustworthiness score. Note that this is not the entire text as this article was very long and had to be truncated for this example image. Some words have been censored in the figure due to offensive language. The highlighted portions show specific textual features that likely affected its trustworthiness score based on the dimension vocabulary matches in the text, thereby the reasons why this article text was given a low trustworthiness score include:

- Very high use of exaggerated language.
- High use of persuasive language and some sentiment.
- No use of statistics.
- No referencing of previous articles, verification of claims, or logical coherence.
- No context or inclusion of multiple perspectives.
- Some use of named entities.
- A positive is that the article title is not clickbait and is objective.

Score: 75
Article Title: The allegations against Daily Mail's parent company Associated Newspapers

Seven prominent individuals have brought legal claims alleging widespread illegal behaviour by individuals working for the Daily Mail and the Mail on Sunday between 1993 and 2018. The Mail's parent company strongly denies all the allegations and is seeking to stop the cases going to trial, arguing the individuals have waited too long to start legal proceedings and are relying on material provided by the Mail on a confidential basis to the Leveson inquiry into the British media industry. On Monday night, a spokesperson for Associated Newspapers said that while "Mail's admiration for Baroness Lawrence remains undimmed, we are profoundly saddened that she has been persuaded to bring this case". They added: "The Mail remains hugely proud of its pivotal role in campaigning for justice for Stephen Lawrence. Its famous 'Murderers' front page triggered the Macpherson report." They also highlighted how one of the private investigators cited by Lawrence has since provided a sworn statement that he did not carry out any illegal work for the Mail or Mail on Sunday. Although the seven individuals' cases are being dealt with collectively, each claim makes distinct allegations of illegal behaviour at Associated Newspapers, the parent company of the Daily Mail and Mail on Sunday – which are strenuously denied by the publisher. This is what they allege. Doreen Lawrence's allegations are particularly damning for the Daily Mail. The mother of murdered schoolboy Stephen Lawrence, whose racially motivated killing shocked Britain, had long been seen as an ally of the Daily Mail and its former editor Paul Dacre. She now alleges that while the paper was publicly campaigning to bring her son's killers to justice – culminating in the newspaper's famous front page accusing a group of men of being murderers – the Mail was also relying on private investigators to dig dirt on her. Lawrence alleges journalists instructed private investigators to conduct illegal interception of her voicemail messages, tapping of her landline, "blagging" of personal records, the monitoring of her bank accounts and phone bills, covert electronic surveillance and corrupt payments to serving Metropolitan police officers working on the murder investigations. Lawrence says she was targeted from "at least as early as 1993 (the year of Stephen's murder) until 2007". She identifies four "unlawful articles".

Figure 3.2: Example of Article Text with High Trustworthiness Score

Figure 3.2 shows an example of an article text from the training dataset that had a high trustworthiness score. Note that this is not the entire text as this article was very long and had to be truncated for this example image. The highlighted portions show specific textual features that likely affected its trustworthiness score based on the dimension vocabulary matches in the text, thereby the reasons why this article text was given a high trustworthiness score include:

- No use of sentiment nor exaggerated language.
- Moderate use of statistics and numerical figures.
- High use of named entities, context, and multiple perspectives.
- No clickbait title.

- High use of verification of claims.
- A negative is there is little to no logical coherence.

3.3 Data Collection

The news article dataset was compiled using various datasets and subsets extracted from Kaggle and GitHub[1] [47], [62]. Preferred facets of these separate datasets to make compilation easier included the dataset having the fields of article origin (i.e., publication, URL, domain), article title, and article text, and that the dataset was in a .csv file for reading in by the Pandas dataframe package in Python. The article title and article text were the most important fields as these contained the necessary NLP dimensions to extract for analysis and prediction. The article origin was used for reference purposes, however when creating the proposed data tool the ultimate objective is for the tool to detect misinformation requiring as few fields as possible and to make predictions based only on the text and not on external factors of misinformation that include article origin, article author, etc. If the tool required additional fields, it should consequently be considered that potential users of the tool may not have or may have difficulty retrieving that information, thereby leading to incomplete or inaccurate predictions from the model.

These are the following sources from where the data originate:

- Tinker Air Force Base Supply Chain Risk Management project dataset

This data was extracted from work on the TAFB SCRM project with the Data Institute for Societal Challenges (DISC). It contains webscraped news articles and blog posts.

- `https://www.kaggle.com/datasets/snapcrack/all-the-news/data`

[1] See also the following URL sources:
`https://www.kaggle.com/datasets/snapcrack/all-the-news/data`,
`https://github.com/several27/FakeNewsCorpus`,
`https://github.com/pmacinec/fake-news-datasets`

This data was from the "All the news" Kaggle dataset and contains news articles from a variety of mainstream and independent news sources. Figure 3.3 shows a bar graph depicting the counts of news articles from the article publication origins featured in the dataset.

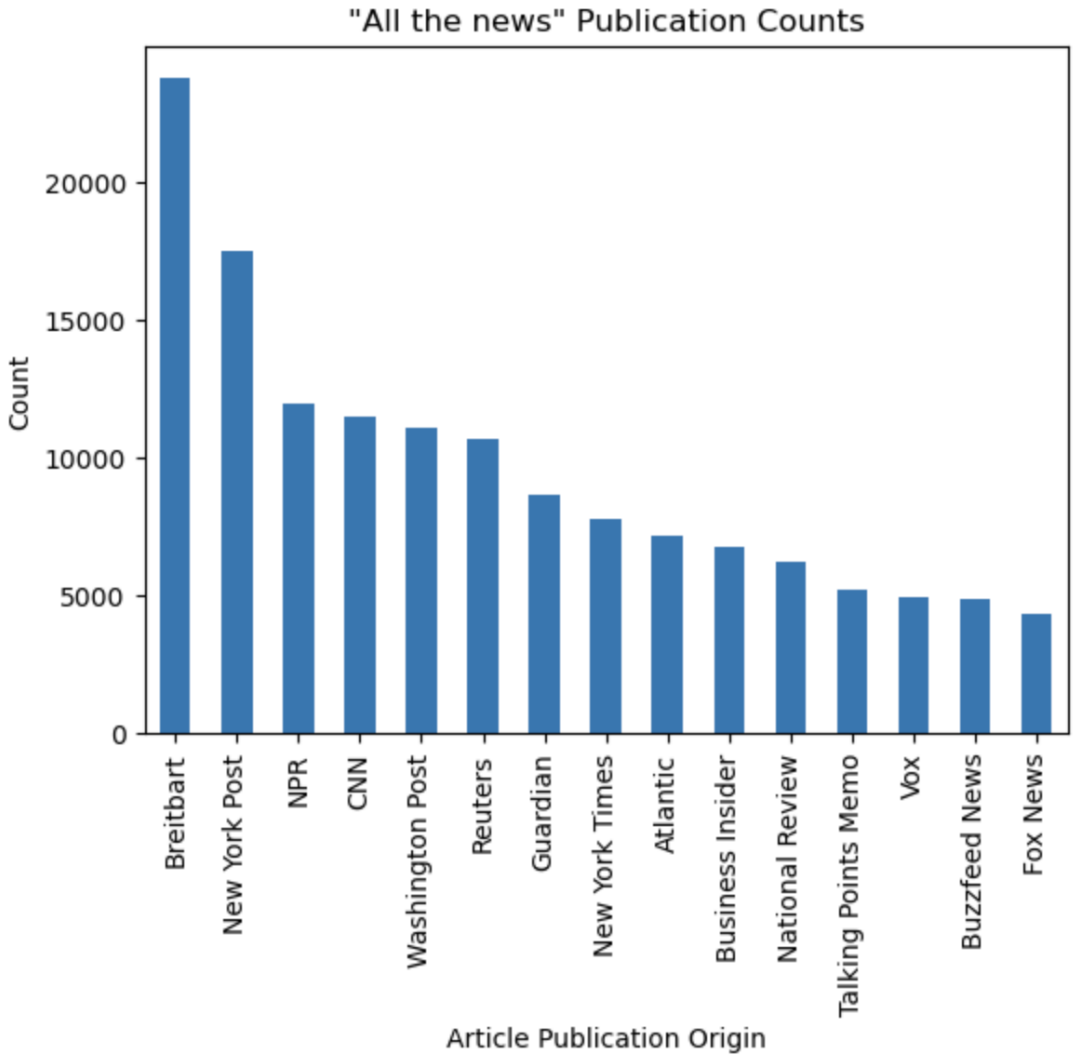


Figure 3.3: Bar Graph of "All the news" Publications

- `https://github.com/qwerfdsaplking/MC-Fake`

This data was from a GitHub dataset that contained news articles from a variety of topics (politics, entertainment, health, COVID-19, and the war in Syria) [47].

- `https://github.com/several27/FakeNewsCorpus`

This data was from a GitHub dataset that contained mostly independent news articles that were categorized as satire, extreme bias, conspiracy theory, hate news, etc. However, only the 250-size open-source sample was used as the entire dataset required an unarchiver app to read the actual larger version of the data.

- `https://github.com/pmacinec/fake-news-datasets`

Multiple datasets can be found in this GitHub repository, however the ones that were chosen were called "fake_news_detection_kaggle" and "getting_real_about_fake_news_kaggle", and these datasets originated from Kaggle. The first dataset listed contains a variety of news articles that contain both real and fake news and the second dataset contains fake news articles that were extracted using the webhose.io API.

- `https://github.com/Gautamshahi/FakeCovid`

This data contains news articles related to COVID-19, a popular topic for fake news detection modeling [62].

3.4 Data Preprocessing

Although the automatic provision of the fields of article origin, article title, and article text in the .csv datasets did assist greatly in preparing the training dataset, many steps had to be taken to preprocess the data.

The separate .csv datasets were loaded into a Jupyter notebook to make it easily accessible to view the various stages of the datasets as they were preprocessed. Even though JupyterLab was initially used for programming during the course of this thesis, Google Colab ended up being used for the majority of coding as this utilized RAM and GPU resources from the cloud rather than resources from the local machine. The following steps were then taken on each dataset before the cleaned dataset was exported to a .csv. First, if the dataset was very large (50,000 articles were needed in total), then random subsets were taken of those datasets to

ensure consistency of the data. Any observations that had a null article origin, article title, or article text were removed entirely as having a nonexistent field would cause issues for the model to make predictions. Unlike preprocessing numerical data where null values can be imputed using the field distribution's mean to retain information, the loss of any fields when it comes to textual data automatically signifies that entire observation cannot be used as it is impossible to impute text. News articles that were not in the English language were removed as well; since the NLP dimensions will be embedded using certain vocabularies in English, vocabularies in foreign languages would require a lot of time and difficulty to compile, and could require the cost of a translation API. Note that this would impose the limitation that the proposed tool would not be functional using foreign language news articles as input. Most texts were above 700 characters in length to include as much context as possible for analysis, however there were some texts under 700 characters that were kept. Nonetheless, an objective of this tool is to make predictions on any news article text regardless of length. Some filler texts were removed from the dataset as these were texts that contained generic information, tended to duplicate each other, and did not provide sufficiently lengthy nor diverse context for analysis.

Finally, as these preprocessing techniques were used on each of the separated datasets and exported as cleaned datasets to .csv files, these cleaned files were then imported back into a Google Colab notebook and merged together using the same column field names "Article Origin", "Article Title", and "Article Text". This joining created a training dataset of 50,092 news articles. In addition, \n characters were removed from the article texts as many of these texts contained these characters, which would be unnecessary for analysis using LLMs. However, most of the texts did not undergo further preprocessing as since LLMs take as much context as possible for analysis and do not require common text preprocessing methods such as lemmatization and stopword removal, it was important to retain as much of the context of the news articles as possible for accurate predictions from the LLMs.

3.5 Exploratory Data Analysis

Looking at the finalized starting dataset, exploratory data analysis was performed on the fields in the dataset and this initial analysis provided some useful insights into the data that will be heavily analyzed during the course of this thesis. There was also exploratory analysis conducted on the NLP dimensions, which helped determine how much each dimension would be weighted for the trustworthiness scoring.

3.5.1 Article Text

First, the cleaned article texts were examined based on length (in characters and words) to determine the approximate distribution of the text length. Statistical summaries and histograms with kernel density curves were made to illustrate the distributions.

```
count     50092.000000
mean       4055.875329
std        4530.083745
min         326.000000
25%        1682.000000
50%        2946.000000
75%        5028.250000
max      152708.000000
Name: Text Length (Characters), dtype: float64
```

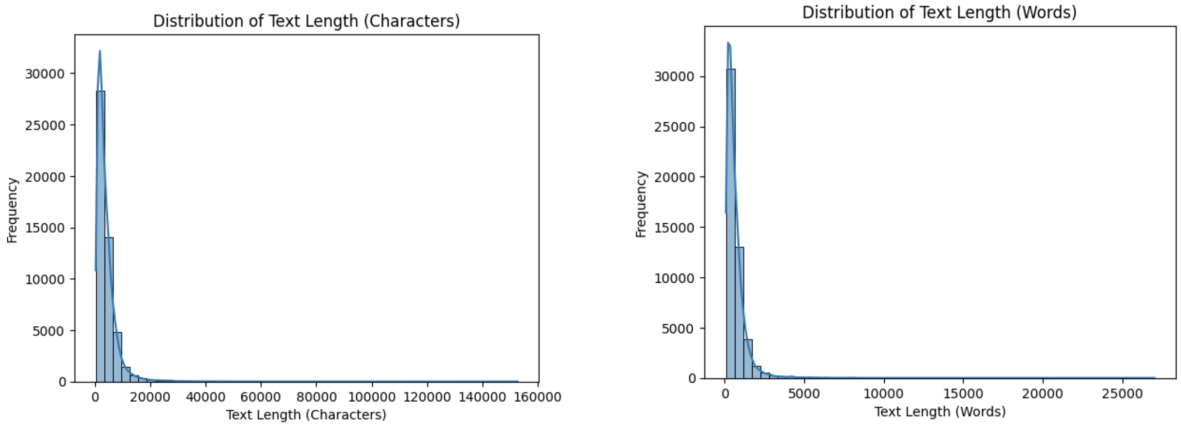
(a) Article Text Length (Characters)

```
count    50092.000000
mean       668.423141
std        761.091305
min         80.000000
25%        276.000000
50%        485.000000
75%        826.000000
max      27013.000000
Name: Text Length (Words), dtype: float64
```

(b) Article Text Length (Words)

Figure 3.4: Statistical Summaries

The statistical summaries show that the median character length of a news article in the training dataset is 2,946 characters. In words, that is equivalent to 485 words. The mean word length of an article text is approximately 668 words, and 25% of all articles in the training dataset are 826 words or above in length.



(a) Article Text Length (Characters)

(b) Article Text Length (Words)

Figure 3.5: Histograms with Kernel Density Curves

Figure 3.5 shows the distributions of the article text lengths, in characters and words, as being heavily skewed right. This is potentially due to not many news articles that exist that are thousands of words in length (the tail in the word length distribution seems to disappear at around 5000 words). Despite the distributions being skewed right as if the articles appear to be short in length, the statistical summaries show that there is indeed a large portion of articles that are rather lengthy.

Therefore, since at least half of the articles in the training dataset are above the maximum input length of 512 tokens, or about 400 words, for a Hugging Face LLM, there is a more emphasized need for including other features in the proposed data tool that can evaluate the article texts in their entirety, going beyond the limitation of the restricted number of input tokens for an LLM.

3.5.2 Article Title

Next, the article title field was analyzed. Analysis was emphasized on word length rather than character length as unlike the article texts the article titles were not filtered out based on a certain character length. Thus, the length of the article titles did not matter as long as the text was not null in the dataset.

```
count    50092.000000
mean        10.381837
std          4.073499
min          1.000000
25%          8.000000
50%         10.000000
75%         12.000000
max        158.000000
Name: Title Length (Words), dtype: float64
```

Figure 3.6: Article Title Length Statistical Summary

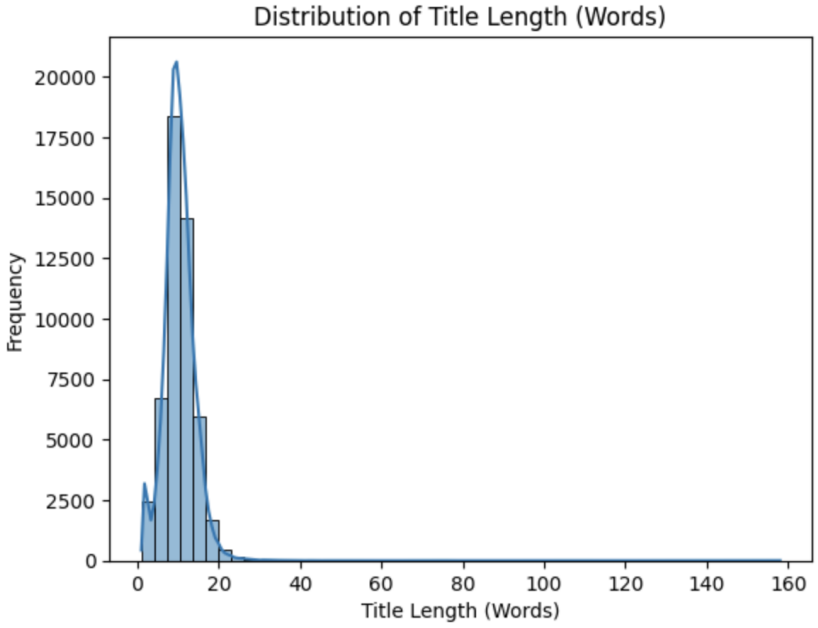


Figure 3.7: Article Title Length Histogram

Figures 3.6 and 3.7 show that the minimum number of words in an article title was 1 (therefore no null or blank values in the article title field per the intended preprocessing), and very few article titles had fewer than 4 words. The mean and median number of words in an article title were around 10, and the maximum number of words was 158. The distribution of the article title length was still skewed right, however not as skewed as the distribution of the article text length. Based on the respective distributions of article text length and article title length, the overall average of an article title length is very short compared to

the article text length. Therefore, there should still be more emphasis on analysis of the comprehensive article text, in addition to the small but substantial role that the article title plays in misinformation detection.

3.5.3 Article Origin

The final field in the dataset that was explored was the article origin field. Since the article origin field was a mixture of article publication names, article URLs, and article domains, the initial number of unique article origins was 27,486. This was primarily because each article URL was unique and had no duplicates, whereas some article publications in the sample format "Fox News" had multiple articles duplicating this publication origin. After simplifying most of the article URLs and domains under singular publication origins to determine which article origins had the highest counts of articles, there were 1,555 simplified unique article origins. Note that some of these were still unique URLs from the same publication source, however there were very few of these observations remaining.

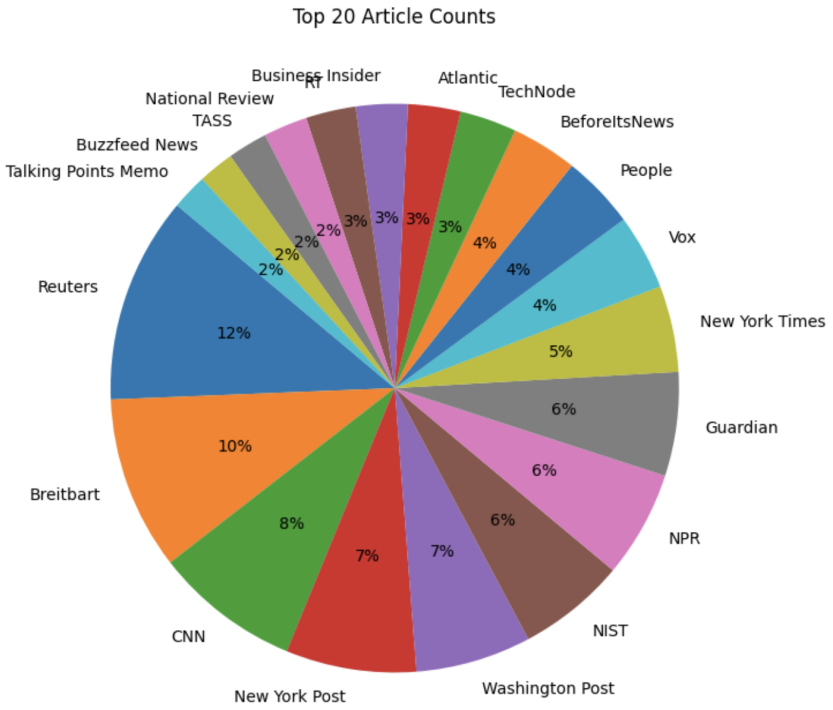


Figure 3.8: Top 20 Article Origins by Count

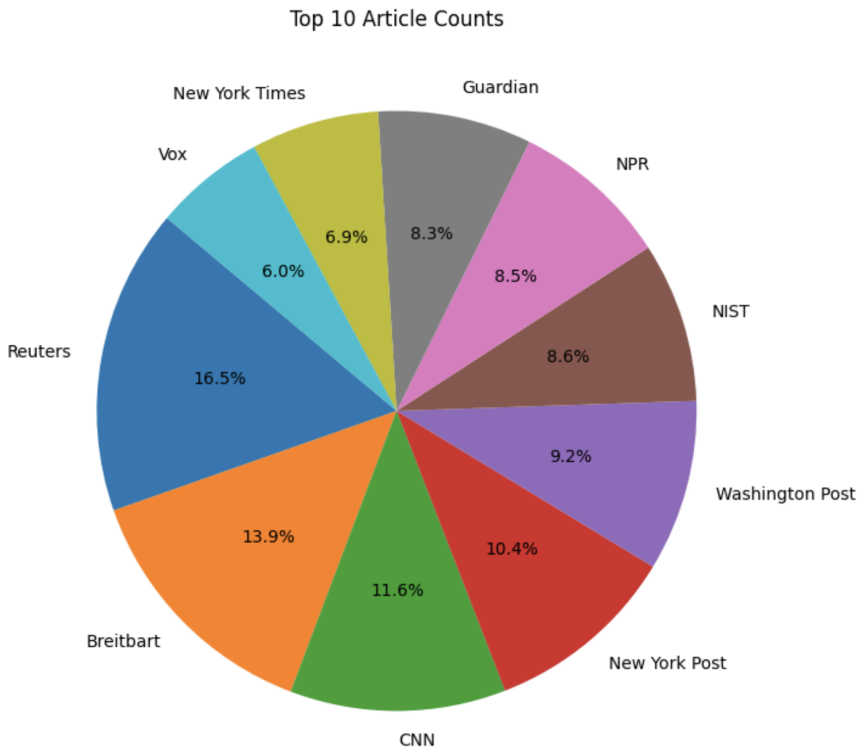
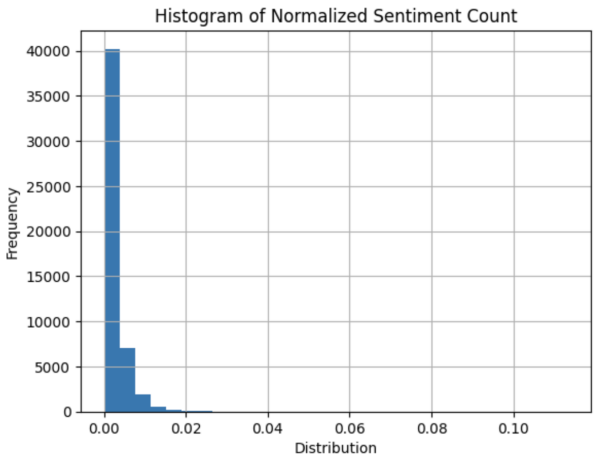


Figure 3.9: Top 10 Article Origins by Count

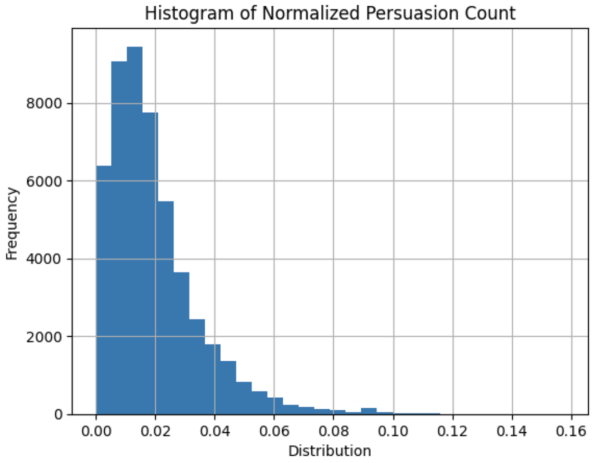
Figures 3.8 and 3.9 depict the top 20 and top 10 article origins by count, and the percentages that are represented in the pie charts are compared to the rest of the top 20 or 10. The top 10 article origins comprised 19,817 articles, or 39.5% of the articles in the training dataset. The top 20 article origins comprised 27,828 articles, or 55.5% of the articles in the training dataset. As for the remainder of the dataset, there are still 1,535 unique article origins that comprise the last 45.5% of the articles. Other notable article origins included Reddit, Microsoft Blogs, Disclose.tv (a far-right publication based in Germany), Fox News, various cybersecurity blogs, and various celebrity news outlets. The exploratory analysis of the article origins has shown that besides mainstream media, there is a large diversity of article origins in the dataset that is sufficiently suitable to train the LLMs to make predictions for a wide variety of news articles.

3.5.4 Natural Language Processing Dimensions

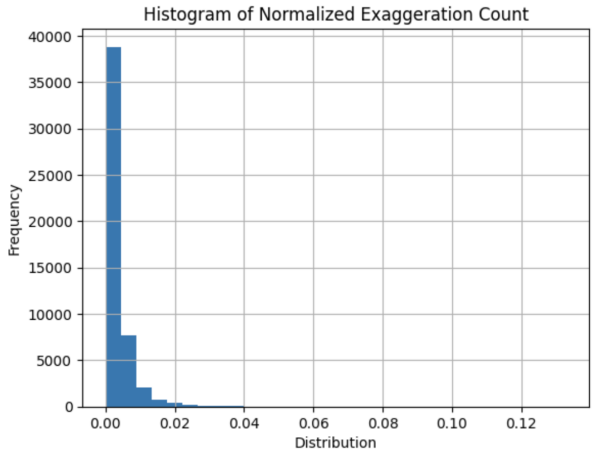
Exploratory data analysis was done on the 13 NLP dimensions that were generated for the articles in the dataset, and the distributions for these articles were analyzed to determine which weights would be assigned to the dimensions to calculate the overall trustworthiness scores for the articles.



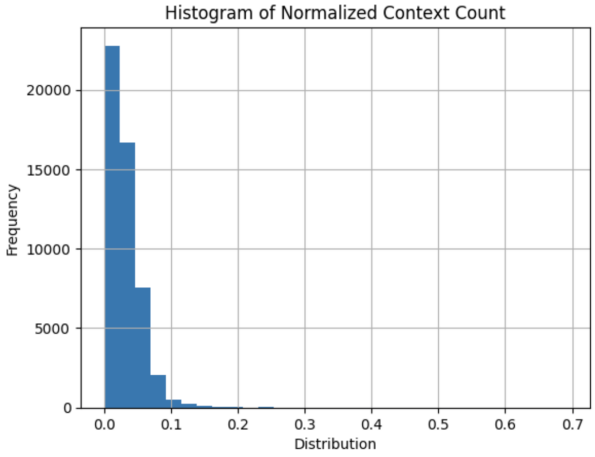
(a) Sentiment



(b) Persuasion

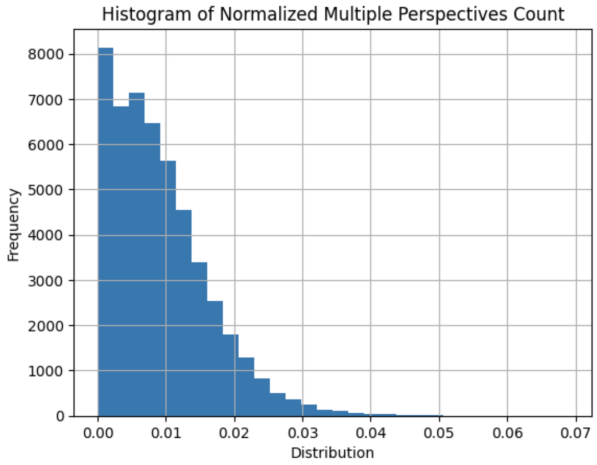


(c) Exaggeration

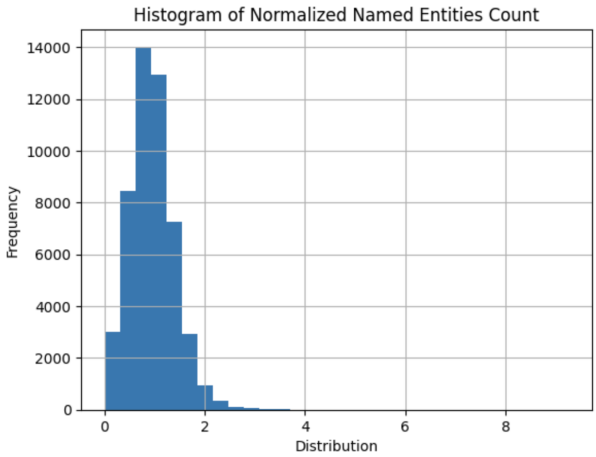


(d) Context

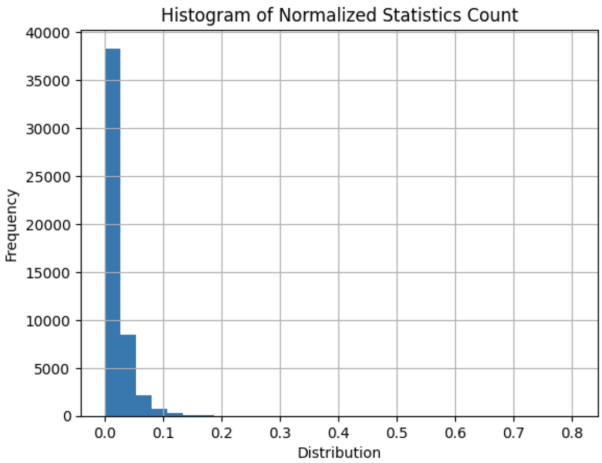
Figure 3.10: NLP Dimension Distributions (1-4)



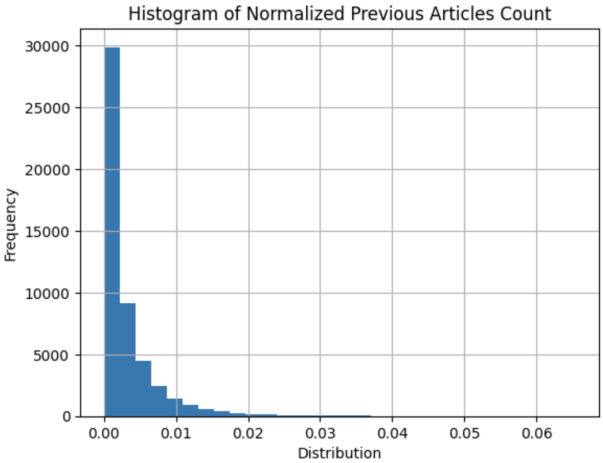
(a) Multiple Perspectives



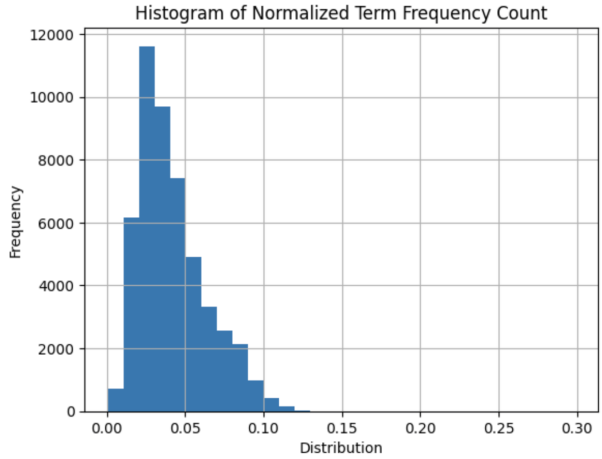
(b) Named Entities



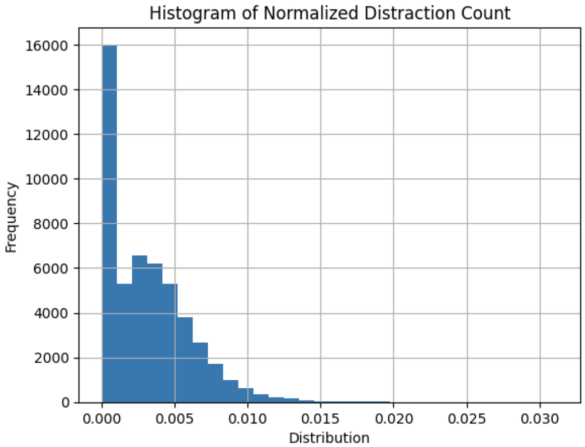
(c) Statistics



(d) Referencing of Previous Articles

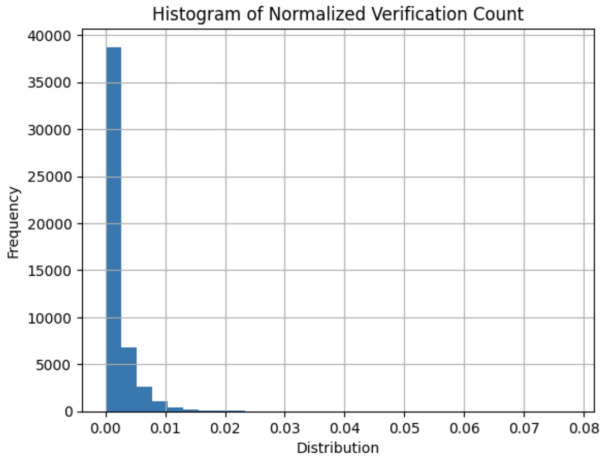


(e) Term Frequency

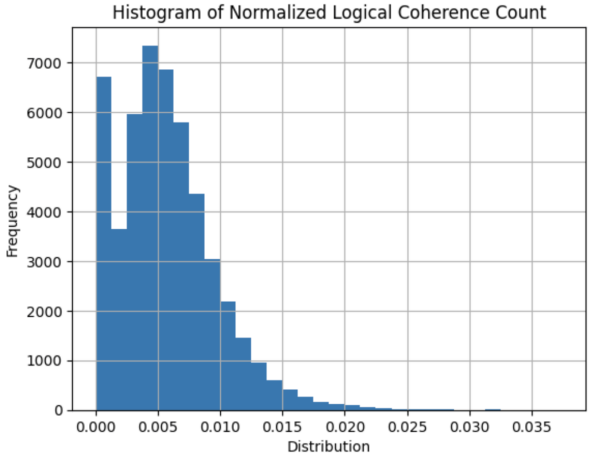


(f) Distraction

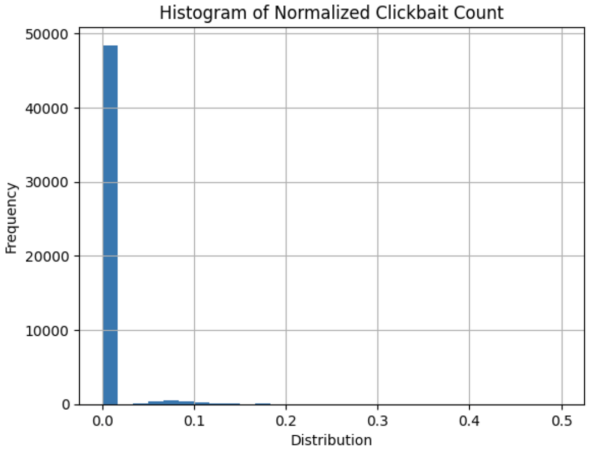
Figure 3.11: NLP Dimension Distributions (5-10)



(a) Verification of Claims



(b) Logical Coherence



(c) Clickbait Title

Figure 3.12: NLP Dimension Distributions (11-13)

As seen in the figures above, some of the dimensions such as sentiment and exaggeration do not appear to have much spread, however since these dimensions are the most indicative of misinformation these were assigned weights of 10. The use of statistics dimension is very skewed, however its center of spread is greater than the other dimensions' distributions (aside from named entities). Although the dimension of logical coherence appears to have a smaller center of spread compared to other distributions, there was more variability in the data, therefore it is a dimension that can have a significant impact on determining misinformation. The dimensions of referencing previous articles, distraction, and verification of claims did not appear to have as much spread nor a greater center of spread compared to other distributions, therefore this is why they were weighted less (however, verification of claims was still crucial enough to be weighted more since it serves as an alternative to fact-checking). Finally, the dimension of clickbait title, based on its distribution spread, appeared to have the least impact in determining misinformation, which is why it was given a weight of 2. It could be considered in the future to increase the weight for this dimension as it can plausibly be a significant indicator for biased news, however the initial exploratory analysis shows insufficient evidence to support increasing the weight for that dimension. As shown in Figure 3.12(c), the vast majority of articles do not have clickbait titles even though some articles may be very biased in their texts, and increasing the weight of the clickbait title dimension would require a more extensive vocabulary for quantifying this dimension.

3.6 Data Labeling

Manually labeling the 50,092 news articles in the training dataset based on the proposed 13 NLP dimensions would have been an extremely difficult and time-consuming process. Thus, a more automated approach to labeling the articles had to be devised. A vocabulary list was made for each NLP dimension (see Chapter 10), with the exceptions of the named entities and term frequency dimensions. These vocabulary lists contained words and punctuation

that would generally be indicative of the dimensions. The vocabulary for the named entities dimension would be extracted using the spaCy package in Python, and the term frequency dimension would not require a vocabulary list, but rather the TF-IDF vectorizer in Python. This would vectorize all of the words in a news article text based on term frequency in relation to the entire corpus of news article texts.

Next, using the NLTK tokenizer in Python, the article texts would be iterated through and the number of times that a token in the dimension vocabulary appeared in an article text would be counted. Then the dimension presence count would be divided by the length of the tokenized text to normalize the count based on the length of the article, quantifying the NLP dimension. In Figure 3.11(b), the normalized count for the named entities dimension can be over 1 because the length of the named entities list extracted from the article text would be divided by the tokenized length.

Finally, taking the distributions of the normalized counts for each dimension, percentiles were calculated to determine which thresholds in the distribution would mean a certain number of points in the dimension weight to be allocated. For example, if the normalized persuasion count was less than or equal to 0.004, the full 10 points would be given for the dimension as this number indicates little to no persuasion present in the article text. As a contrary example, if the normalized logical coherence count was greater than 0.0125, the full 10 points would be awarded since this threshold indicates that there is a high presence of logical coherence in the article text. After the individual dimension scores were allocated, these scores were summed up to calculate the cumulative trustworthiness score on a scale from 0 to 100. Figures 3.13 and 3.14 show the distribution of the final score labels.

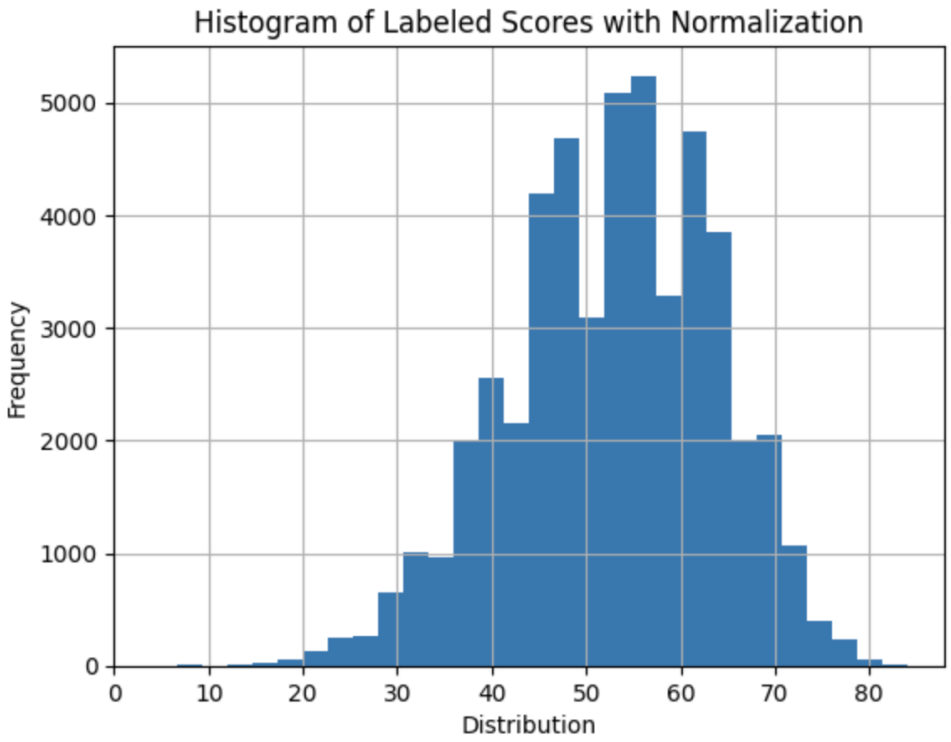


Figure 3.13: Distribution of Score Labels

```
count    50092.000000
mean        52.618941
std         10.885496
min          4.000000
25%         45.000000
50%         53.000000
75%         61.000000
max         84.000000
Name: New Labeled Scores (Normalization), dtype: float64
```

Figure 3.14: Statistical Summary of Score Labels

The distribution of the score labels follows a Gaussian, or normal, distribution, with a center of spread (the mean) at around 52. 50% of the score labels are below 53, the minimum automated score label is 4, and the maximum automated score label is 84.

Some data exploration was also done by dividing the labeled news articles into the categories "Likely Trustworthy" and "Likely Untrustworthy". If the labeled cumulative trustworthiness

score was below a 50, it would be classified as "Likely Untrustworthy", and if the cumulative score was above a 50, it would be classified as "Likely Trustworthy". This classification threshold of 50 would likewise be utilized for classification tasks later on in the thesis. 18,964 articles were "Likely Untrustworthy" and 31,128 articles were "Likely Trustworthy". In addition, 17,435 articles out of the training dataset had a trustworthiness score between 45 and 55, the closest interval in the middle that could determine which articles would be trustworthy and which ones would not be. While many articles were close to the trustworthiness score threshold of 50, these articles tended to have little to no context, logical coherence, inclusion of multiple perspectives, nor use of statistics, meaning a large deduction from the trustworthiness score that could determine, by a few points, whether an article could be trustworthy or not.

3.7 Model Training

The following steps discussed in this section were taken for the data preparation and training for all Hugging Face LLMs. First, the quantified NLP dimensions for the given article text were concatenated with the text to improve the accuracy of the LLM. The dataset would then be split up into 70% training, 15% validation, and 15% testing. All LLMs were imported from the Transformers package in Python and were trained using the TensorFlow package. Although the PyTorch package, which is also commonly used for training LLMs, was tested, the models appeared to train slower and less accurately using PyTorch compared to using TensorFlow.

The training, validation, and testing sets were tokenized using the imported LLM tokenizers and converted into input IDs, or token indices that are numerical representations of the token sequences that will be accepted by the LLM. The input IDs would then be compiled into a Keras Input layer for the model. A Keras Dense layer creating the regression head for the model was also created to fine-tune the model. After the overall model with the layers

was initialized, the model would be compiled with the Adam optimizer with a learning rate of $1e^{-5}$, mean squared error (MSE) as the loss function, and mean absolute error (MAE) as an additional evaluation metric. The batch size for training all LLMs was 32, and Keras early stopping was implemented to track and save the epoch where the model had the least validation loss; the patience was set at 2 epochs until no improvement in decreasing the validation loss was evident. The combination of the Adam optimizer, $1e^{-5}$ learning rate, and batch size of 32 were consistently chosen to have fast and efficient training of the LLMs while keeping training stable so as to not have the loss function converge too quickly. For testing evaluation metrics, MSE, MAE, Pearson Correlation Coefficient, Spearman Rank Correlation Coefficient, and R^2 were used. All model weights were saved to be later imported for making trustworthiness score predictions.

3.7.1 BERT

The imported tokenizer and pre-trained model from the Transformers package were "bert-base-uncased". The maximum input token sequence length that was used by the model was 176; maximum input sequences higher than the designated token length led to GPU overload in Google Colab, and this applied to all LLMs. In addition to input IDs, the BERT model also used attention masks, or encodings that match the length of the input IDs of one text to the input IDs of another text. Attention masks are an optional argument while training LLMs, however they are recommended for controlling variable sequences. The input IDs and attention masks along with the score labels were converted into TensorFlow datasets for input into the BERT model.

```
Model: "model"
_________________________________________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
=================================================================================================
 input_ids (InputLayer)      [(None, 176)]             0         []

 attention_mask (InputLayer  [(None, 176)]             0         []
 )

 tf_bert_model (TFBertModel  TFBaseModelOutputWithPooli  109482240  ['input_ids[0][0]',
 )                           ngAndCrossAttentions(last_  0          'attention_mask[0][0]']
                             hidden_state=(None, 176, 7
                             68),
                              pooler_output=(None, 768)
                             , past_key_values=None, hi
                             dden_states=None, attentio
                             ns=None, cross_attentions=
                             None)

 dense (Dense)               (None, 1)                 769       ['tf_bert_model[0][1]']

=================================================================================================
Total params: 109483009 (417.64 MB)
Trainable params: 109483009 (417.64 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________________________________________
```

Figure 3.15: BERT Model Architecture Summary

3.7.2 RoBERTa

The imported tokenizer and pre-trained model from the Transformers package were "roberta-base". The training of the RoBERTa model also utilized attention masks. The maximum sequence length determined was 160 tokens. As the BERT model only used the regression head for output, the RoBERTa model instead used the regression head with CLS (classifier) tokens. Before this model was trained, the tokenized training, validation, and testing sets had the input IDs and attention masks extracted and converted into NumPy arrays, and the score labels were also converted into a NumPy array for training to function.

```
Model: "model"
_________________________________________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
=================================================================================================
 input_ids (InputLayer)      [(None, 160)]             0         []

 attention_mask (InputLayer  [(None, 160)]             0         []
 )

 tf_roberta_model (TFRobert  TFBaseModelOutputWithPooli  124645632  ['input_ids[0][0]',
 aModel)                     ngAndCrossAttentions(last_            'attention_mask[0][0]']
                             hidden_state=(None, 160, 7
                             68),
                              pooler_output=(None, 768)
                             , past_key_values=None, hi
                             dden_states=None, attentio
                             ns=None, cross_attentions=
                             None)

 tf.__operators__.getitem (  (None, 768)               0         ['tf_roberta_model[0][0]']
 SlicingOpLambda)

 dense (Dense)               (None, 1)                 769       ['tf.__operators__.getitem[0][
                                                                 0]']

=================================================================================================
Total params: 124646401 (475.49 MB)
Trainable params: 124646401 (475.49 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________________________________________
```

Figure 3.16: RoBERTa Model Architecture Summary

3.7.3 DistilBERT

The imported tokenizer and pre-trained model from the Transformers package were "distilbert-base-uncased". The determined maximum sequence length was 300 tokens, which can be explained by DistilBERT being a much lighter version of BERT that can retain most of BERT's performance. No attention masks were used in the training of the DistilBERT model, and the regression head contained a pooling output.

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_ids (InputLayer)      [(None, 300)]             0

 tf_distil_bert_model (TFDi  TFBaseModelOutput(last_   66362880
 stilBertModel)              hidden_state=(None, 300
                             , 768),
                              hidden_states=None, at
                             tentions=None)

 tf.__operators__.getitem (  (None, 768)               0
 SlicingOpLambda)

 dense (Dense)               (None, 1)                 769

=================================================================
Total params: 66363649 (253.16 MB)
Trainable params: 66363649 (253.16 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 3.17: DistilBERT Model Architecture Summary

3.8 Results and Interpretation

The model trainings, without the addition of the NLP dimensions as features, produced correlation coefficients of around 0.73 for all LLMs. This would be equivalent to an R^2, or goodness of fit, of 0.53, which would not make the models even fairly accurate in determining the trustworthiness scores. The figures below show that all 3 LLMs with the additions of the NLP dimensions had very good accuracy and goodness of fit, and the RoBERTa model especially had the overall lowest MSE and the highest overall correlation. However, some drawbacks of the RoBERTa model included the model accepting the least amount of input tokens and requiring the longest training time. The addition of the NLP dimensions significantly improved the accuracy of the models.

Model	Number of Epochs	Maximum Input Tokens	Training Loss (MSE)	Training MAE
BERT	8	176	4.4780	1.4970
RoBERTa	17	160	1.6063	0.9956
DistilBERT	12	300	3.6656	1.4519

Figure 3.18: Model Training Results

Model	Validation Loss (MSE)	Validation MAE	Testing Loss (MSE)	Testing MAE
BERT	2.8110	1.1227	2.7812	1.1202
RoBERTa	2.2895	1.1895	2.2804	1.2012
DistilBERT	2.3013	1.1119	2.2903	1.1128

Figure 3.19: Model Validation and Testing Results

Model	Pearson Corr. Coefficient	Spearman Rank Corr. Coefficient	R^2 (Pearson Coefficient)	R^2 (Spearman Rank Coefficient)
BERT	0.9884	0.9942	0.9770	0.9885
RoBERTa	0.9952	0.9966	0.9904	0.9932
DistilBERT	0.9903	0.9928	0.9807	0.9856

Figure 3.20: Model Correlation Coefficient Results

Despite these promising results, statistical significance tests needed to be performed between combinations of the model scores and the true scores to validate if the RoBERTa model would still be the best choice for the proposed data tool. Using the same subset of 1000 random observations that will be used for the unsupervised learning methods and AI model evaluation, the real scores of those articles were compared with the predictions that the BERT, RoBERTa, and DistilBERT models would make on that same data.

T-test	T-statistic	P-value
BERT and RoBERTa	-37.7539	6.5889e-236
RoBERTa and DistilBERT	-6.0106	2.1912e-9
BERT and DistilBERT	-40.5685	5.3107e-263
Real Scores and BERT	17.7735	9.9483e-66
Real Scores and RoBERTa	-3.2059	0.0014
Real Scores and DistilBERT	-6.7062	2.5910e-11
ANOVA	**F-statistic**	**P-value**
BERT, RoBERTa, and DistilBERT	1000.9154	0

Figure 3.21: Statistical Significance Test Results

Using T-tests and ANOVA to compare the means of these distributions, the mean of the real score subset data was very close to the mean of the real scores of the overall dataset, which means that these distributions would likely be normal. Every test showed that the means of the sets of prediction scores and real scores were statistically significant as every p-value was less than the designated value of α of 0.05. The ANOVA test showed that there was a significant difference between the 3 model prediction score sets. The T-test comparison that showed the least difference was between the real scores and the RoBERTa prediction scores, and the T-test comparison that showed the greatest difference was between the BERT prediction scores and the DistilBERT prediction scores.

Further validation of the results was done by using a subset of the WELFake Kaggle dataset (`https://www.kaggle.com/datasets/saurabhshahane/fake-news-classification`) [75]. Using this subset, the scores that would be calculated using the NLP dimension algorithm versus the scores that would be calculated using the RoBERTa model were compared with the labels in the dataset. Note that the labels in the WELFake dataset were binary (0 and 1), so only classification accuracy could be used to compare these results. Although the accuracies were low due to the NLP dimension vocabularies lacking complexity and real-world examples found in the news today (these would likely improve the efficacy in the future), these results showed that the RoBERTa model still performed better than the NLP dimension score calculations. These predictions were also statistically significant, meaning that

the RoBERTa model could still perform better compared to only using the NLP dimension score calculations.

Type of Predictions	Accuracy	Precision	Recall	F1-Score
NLP Dimensions	0.3011	0.2475	0.2070	0.2255
RoBERTa	0.4486	0.4612	0.7267	0.5643

T-statistic	-20.345	P-value	1.180e-83

Figure 3.22: NLP Dimension Predictions vs. LLM Predictions Results

Therefore, the primary conclusion reached based on the model training and analysis is that the RoBERTa model prediction scores follow the distribution and center of the real trustworthiness scores the most, making it the most efficient choice for the proposed data science tool, while BERT would be the least effective choice as it had the lowest accuracy of the three LLMs and the most disparity between its prediction scores and the real scores.

3.9 Model Integration in Proposed Data Tool

To integrate the chosen LLM into the proposed data science tool, the technique of extracting the quantified NLP dimensions and concatenating them with the article texts was first performed to create the inputs for the tool. This tool would require an input dataframe of article texts and titles to generate the predictions effectively. The concatenated texts would be encoded using the RoBERTa tokenizer and converted into input IDs and attention masks to feed into the RoBERTa base model of the tool. Note that the exact model architecture would be replicated as it was during training in order for the model prediction to function properly. The model would then generate the score predictions and in addition to the predictions, the tool would output explainability metrics (i.e., presence of NLP dimension, individual NLP dimension score, named entities list) to explain why the model likely gave the provided score to the article text. Finally, the prediction scores would be stored in a new

column in the input dataframe and the dataframe with the predictions would be exported to the user.

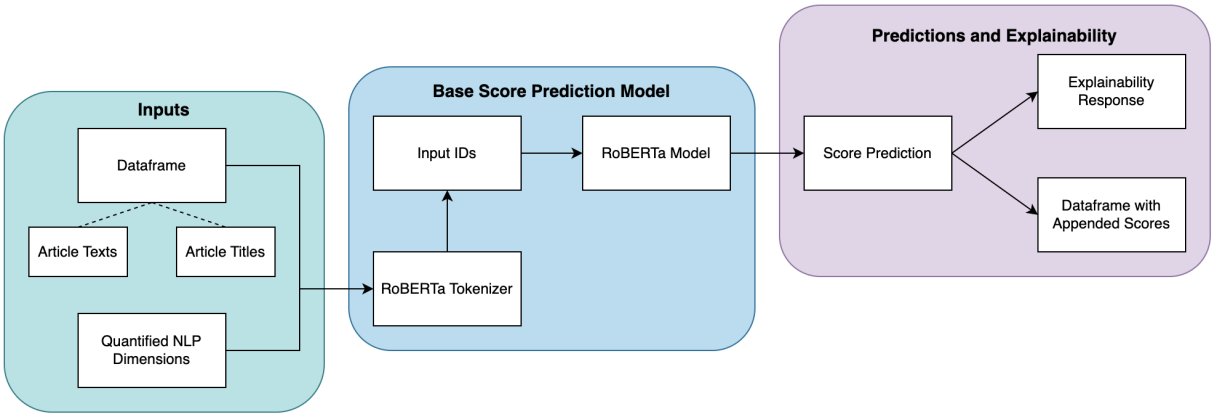


Figure 3.23: Diagram of Proposed Data Tool Structure

```
{
    "trustworthiness_score": 67,
    "dimension_scores": {
        "sentiment": 9,
        "persuasion": 7,
        "exaggeration": 8,
        "context": 4,
        "multiple_perspectives": 6,
        "named_entities": 10,
        "statistics": 2,
        "previous_articles": 1,
        "term_frequency": 3,
        "distraction": 2,
        "claim_verifications": 3,
        "logical_coherence": 10,
        "clickbait_title": 2
    },
    "explainability": {
        "sentiment": "not present",
        "persuasion": "little presence",
        "exaggeration": "little presence",
        "context": "some presence",
        "multiple_perspectives": "some presence",
        "named_entities": "present",
        "statistics": "not present",
        "previous_articles": "some presence",
        "term_frequency": "little presence",
        "distraction": "not present",
        "claim_verifications": "present",
        "logical_coherence": "present",
        "clickbait_title": "not present"
    }
}
```


Figure 3.24: Sample Proposed Data Tool Response

3.10 Summary

This chapter of the thesis explored multiple Hugging Face LLMs and their effectiveness in determining trustworthiness scores for news articles regardless of length. Communication elements of misinformation were converted into quantified NLP dimensions using indicative

vocabularies, and the inclusion of these dimensions into the LLM training significantly improved the accuracy of the LLMs. These NLP dimensions also served as an important feature to comprehensively label and assist in providing explainability for the score predictions.

After training all LLM options, RoBERTa had the best performance as its predictions followed the distribution of the score labels the closest, despite it requiring the least amount of input tokens or context. DistilBERT required the most input tokens and performed fairly well, and BERT overall performed the worst as it had the least accuracy and the most disparity between its predictions and the real scores. Therefore, RoBERTa was chosen to be the base model for the proposed data tool.

Finally, a data science tool was proposed that would serve the purpose of efficiently generating score predictions through the base LLM by holistically evaluating the entire context of the news articles using the NLP dimensions given an input dataframe of articles. This tool would also create responses of explainability metrics that would assist with ensuring trustworthiness in the tool by justifying why the given scores were generated for the news articles. Likewise, score predictions can be stored and exported in dataframes provided by the user.

Despite the evaluation of the fine-tuned supervised learning LLMs, unsupervised learning models and AI models need to be compared to the performances of the LLMs to resolve which of these 3 types of models is most recommended for future research and use for determining misinformation potential accurately.

Chapter 4

Unsupervised Learning Insights and Data Visualizations

4.1 Overview

Unsupervised learning models will be the second type of model explored in this thesis. These models do not require a labeled dataset nor training to make predictions, however they can produce valuable insights about data including which clusters certain data points are a part of. Given this application, these models can have the potential to produce novel revelations about trustworthy and untrustworthy news articles.

The unsupervised learning methods that will be used for this portion of the thesis include:

- Method 1: Anomaly detection in tandem with LDA topic modeling.
- Method 2: Cosine similarity in tandem with t-SNE.

These 2 methods will be performed on a subset of the article dataset. Association rule mining was also experimented with, however it was not be able to be used on the articles due to GPU overload, and the last section of this chapter will briefly discuss related analysis that was still conducted using this unsupervised learning method.

This chapter of the thesis will add to the conclusions reached from Chapter 3 and will attempt to eliminate ambiguity between what makes a news article trustworthy or untrustworthy

through effective data visualizations and insights.

4.2 Data Collection

The following dataset will be used for the unsupervised learning analysis in this chapter:

- 1000 random observations from the training set used for the LLMs

Out of the 50,092 news articles in the training dataset, 1000 randomly extracted articles will be used that follow the same score label distribution as the overall dataset. These observations will be divided into 2 trustworthiness categories based on the score label range that the articles fall into. These same 1000 observations will also be used in Chapter 5 of this thesis for more effective and consistent comparisons.

4.3 Method 1

This approach involved anomaly detection and LDA modeling. The objective of this method was to define an outlier subset of news articles and to distinguish the component differences between the normal observations and anomaly observations between the articles that were labeled "Likely Trustworthy" and "Likely Not Trustworthy". LDA models were also initialized to generate the top keywords and topics of the "Likely Trustworthy" and "Likely Not Trustworthy" article anomalies. If the isolation forest model, which is responsible for determining the anomalies that differ from the normal behavior of the data, would generate anomalies that are trustworthy and not trustworthy articles, then the anomalies would be most indicative of which articles are likely to be trustworthy and which articles are likely to be not trustworthy. Although unsupervised learning methods do not require labeled data, revealing patterns that distinguish trustworthy and untrustworthy news articles could pave the way for unsupervised learning misinformation detection research that will not require labeled data.

4.3.1 Data Preprocessing

For preprocessing related to anomaly detection, the 1000 observations were divided into 2 categories (depending on whether the labeled score was above 50 or below 50 as previously explored in the training dataset in Chapter 3). This data would then be fed into the isolation forest model.

The data preprocessing regarding the LDA modeling involved extracting the anomalies that were labeled "Likely Trustworthy" and the anomalies that were labeled "Likely Not Trustworthy" to initialize the LDA models and generate wordcloud data visualizations for the anomaly categories. All of the cleaned texts were tokenized and set to lowercase, and lists of words and punctuation to exclude from analysis were defined that included insignificant words and punctuation. It was important to ensure that as many prominent words as possible were featured in the topics generated by the LDA model.

4.3.2 Anomaly Detection

The corpus of article texts was converted into a TF-IDF vector matrix, an acceptable input for the isolation forest model. Then, the matrix had its features reduced using truncated single value decomposition; the number of components, or features, to be analyzed was set to 2, therefore the anomalies would be analyzed on a 2-dimensional space. Then, the reduced features were input into the isolation forest model with a contamination (proportion of data that would be selected as outliers) of 0.1, so 10% of the data was chosen as anomalies to have sufficient data from both trustworthiness categories for the anomaly detection analysis. In the dataframe of article texts, anomalies were given a label of -1 to indicate that these were anomalies; 62 articles were "Likely Not Trustworthy" anomalies and 38 articles were "Likely Trustworthy" anomalies. Next, data visualizations were made comparing the anomaly data to the normal data, which will be discussed in the Results and Interpretation subsection.

4.3.3 Latent Dirichlet Allocation Topic Modeling

For each trustworthiness category, the preprocessed article text anomalies were converted into a dictionary and then into a corpus of text using bag-of-words. The LDA model would be initialized using the dictionary, the corpus, and the following parameters:

- Number of topics: 3
- Number of passes: 10
- α: 0.7
- η: 0.7

After the LDA model was trained to perform the topic modeling, 3 keywords were generated for each of the 3 topics specified by the model.

4.3.4 Results and Interpretation

The results of the anomaly detection and LDA modeling, especially through the use of data visualization, revealed some enlightening insights. Figure 4.1 shows the anomaly data using purple dots and the normal data using yellow dots, displayed in a 2-dimensional space that represents the 2 components, or reduced features of the data.

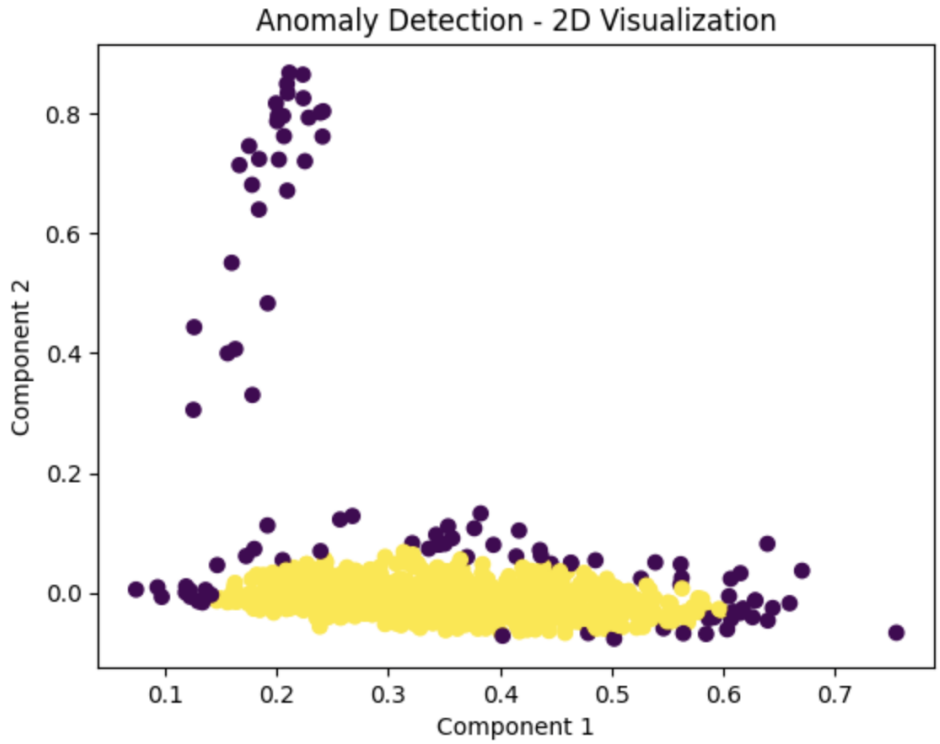
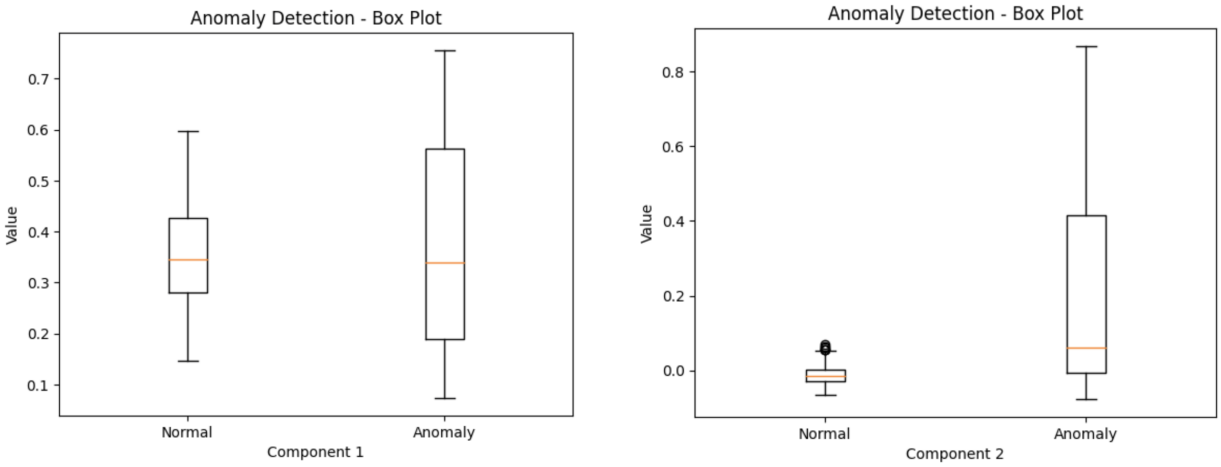


Figure 4.1: 2-Dimensional Visualization of Anomalies

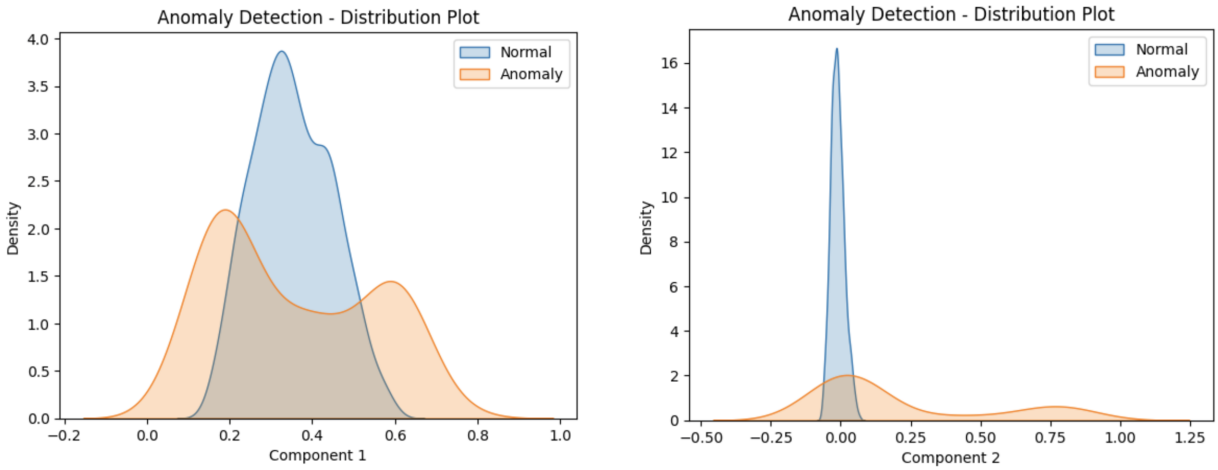


(a) Component 1 (b) Component 2

Figure 4.2: Boxplot Visualizations of Anomalies

Based on the visualizations of the components comparing the normal and anomaly data in Figures 4.1 and 4.2, these show that the normal data appear to be concentrated within the range of 0.15 and 0.6 for the value of Component 1, and concentrated within the range of -0.1

and 0.1 for the value of Component 2. In contrast, the anomaly data tend to have a larger range, with 0.1 to 0.75 for Component 1 and -0.1 to 0.9 for Component 2. The centers of the boxplot distributions for the normal and anomaly data appear to be the same at about 0.35 for Component 1, but the centers of the distributions for Component 2 are more different, with the normal data centered around 0.0 and the anomaly data centered around 0.1.



(a) Component 1 (b) Component 2

Figure 4.3: Distribution Visualizations of Anomalies

Density distribution visualizations likewise provided crucial content for analysis. Figure 4.3 shows the normal data distributions being unimodal for both components while the anomaly distributions for both components are bimodal. Given this insight, additional analysis was done to see which trustworthiness category was more prevalent in the anomaly data distributions given specified thresholds of the component values. It was found that likely trustworthy articles were twice as prevalent compared to likely untrustworthy articles with a Component 1 value greater than 0.5, and that likely not trustworthy articles were three times as prevalent with a Component 1 value less than 0.3. Likely not trustworthy articles were almost 7 times more prevalent with a Component 2 value greater than 0.5, and likely trustworthy articles were almost 2 times more prevalent with a Component 2 value less than 0.05. Therefore, trustworthy anomalies tended to have a higher Component 1 value and a lower Component 2 value, while untrustworthy anomalies tended to have a lower

Component 1 value and a higher Component 2 value.

When using a contamination of 0.05, the results using the initial contamination of 0.1 changed some; regardless of component values, "Not Likely Trustworthy" anomalies were more present within each of the lower and higher value ranges of the distributions. However, "Not Likely Trustworthy" articles were 4 times more prevalent than "Likely Trustworthy" articles in the anomalies determined by the isolation forest model (39 to 11). Thus, while having a lower contamination can lead to one trustworthiness category likely being isolated, a higher contamination can provide further differentiations between the 2 trustworthiness categories through their component values.

```
Topic  1
[('trump', 0.0075849895), ('vulnerability', 0.00691555), ('user', 0.0039968286)]
Topic  2
[('wagner', 0.0059274086), ('prigozhin', 0.0049258703), ('group', 0.0043812166)]
Topic  3
[('court', 0.0028152203), ('state', 0.0025498641), ('senate', 0.0023238908)]
```

Figure 4.4: Topics and Keywords in Likely Trustworthy Anomaly Articles

```
Topic  1
[('sites', 0.011470512), ('nist', 0.006860592), ('links', 0.006362473)]
Topic  2
[('chemistry', 0.0033402452), ('people', 0.0028704773), ('identity', 0.002563864)]
Topic  3
[('people', 0.0060038827), ('bash', 0.0036462797), ('us', 0.0034702562)]
```

Figure 4.5: Topics and Keywords in Likely Not Trustworthy Anomaly Articles

The topics and keywords generated by the LDA models on the likely trustworthy and likely untrustworthy category data showed that articles discussing topics such as former U.S. president Donald Trump, the Wagner Group and Prigozhin in Russia, and the U.S. Senate and court system are likely trustworthy, and articles discussing topics such as NIST websites and links and people and identity are likely not trustworthy.

Wordcloud data visualizations were also created to illustrate the most significant words present in the anomaly data for each trustworthiness category. The article texts were pre-

processed the same way as they were for the LDA modeling, however a counter was also used for counting the number of times a given word appeared in the corpus of text. The wordcloud displayed all significant words that were present at least 10 times in the corpus.

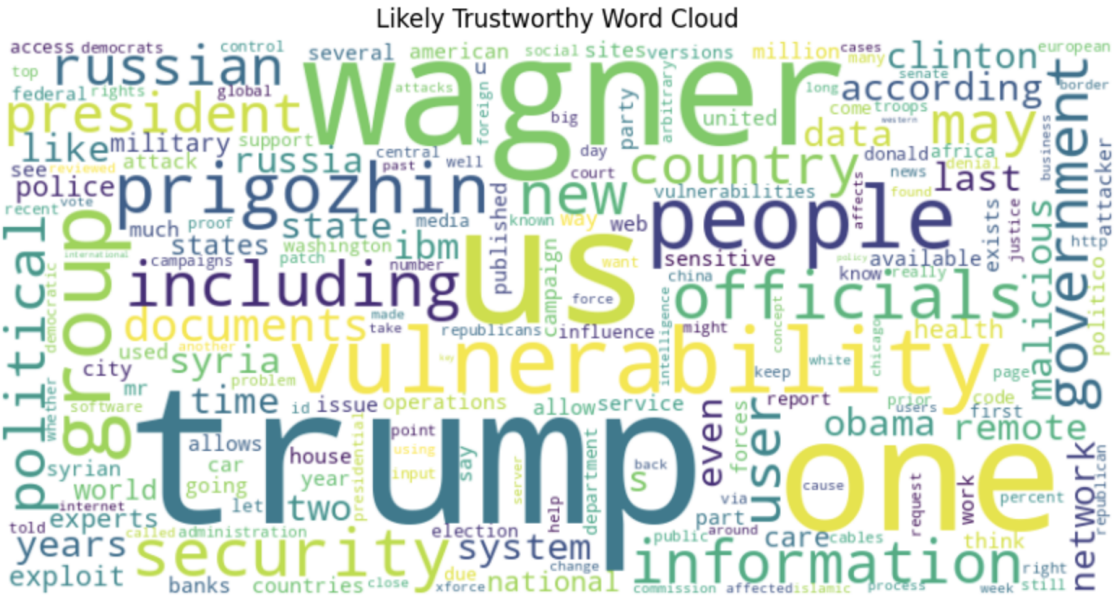


Figure 4.6: Likely Trustworthy Article Words

Figure 4.7: Likely Not Trustworthy Article Words

The wordcloud visualizations contain some of the same words that were shown in the gener-

ated topic keywords from the likely trustworthy and likely untrustworthy articles. In addition, words that indicate articles discussing topics including politics, public officials, former presidential candidate Hillary Clinton, government, the war in Syria, and web vulnerabilities and security are articles that are likely to be trustworthy. Articles that include words such as information, content, bash, endorse, think, and page indicate that these articles are likely to be not trustworthy. There are some words that are present in both wordclouds, which still leaves some ambiguity between trustworthy and untrustworthy news article keywords.

Therefore, based on this unsupervised learning approach, this analysis has shown that certain keywords as generated from the LDA modeling and wordcloud visualizations can provide indications for whether a news article is likely to be trustworthy or untrustworthy based on its topic. However, there can be some overlap between words occurring in both trustworthiness categories. Furthermore, this method has shown that if using dimension reduction through anomaly detection on news articles (set to 2 dimensions only), an article with a higher value for Component 1 was more likely to be trustworthy, and an article with a higher value for Component 2 was more likely to be untrustworthy.

4.4 Method 2

This approach involved cosine similarity and t-SNE. The objectives of this method were to determine if there were distinctions between the cosine similarities of the likely trustworthy news articles and the likely untrustworthy news articles and to create visualizations to see if certain clusters of likely trustworthy news articles and likely untrustworthy news articles tended to separate from each other. Component distributions for each trustworthiness category were also analyzed to see if there were any notable differences. If any significant distinctions would be revealed between the trustworthiness categories, this method would make an effective indicator to conclude which news articles are trustworthy or not in future misinformation detection research.

4.4.1 Data Preprocessing

The same 1000 observations were used and divided into the same trustworthiness categories as in Method 1. Additionally, the categorized news articles were combined and preprocessed using NLTK tokenization, setting all tokens to lowercase, and removing stopwords and punctuation. This cleaned data would then be fed into the Word2Vec model that would then be used to calculate the cosine similarities between the news articles in vector form.

4.4.2 Cosine Similarity

The news articles texts were converted into word embeddings, or vectors, using a Word2Vec model in the Gensim package, and each vector was set to a size of 200. Then the model vocabulary was constructed and the model trained. Document vectors for the articles were generated by iterating through the processed articles to create a single vector for each article. The cosine similarities comparing the document vectors were calculated and put into vector form.

4.4.3 t-Distributed Stochastic Neighbor Embedding

Taking the document vectors, these were put into a NumPy array and any remaining missing document indices were dropped. This array was then transformed into embedded vectors using t-SNE with the following hyperparameters:

- n_components: 2
- perplexity: 20
- random_state: 42

Therefore, the t-SNE analysis will be done on a 2-dimensional scale.

4.4.4 Results and Interpretation

Using the cosine similarity vectors generated, the averages of the vectors were calculated and a violin plot was made comparing the average cosine similarities of each trustworthiness category. Figure 4.8 shows that the distributions of the average cosine similarity for each trustworthiness category were approximately the same, with the exception of differences between the amounts of outliers.

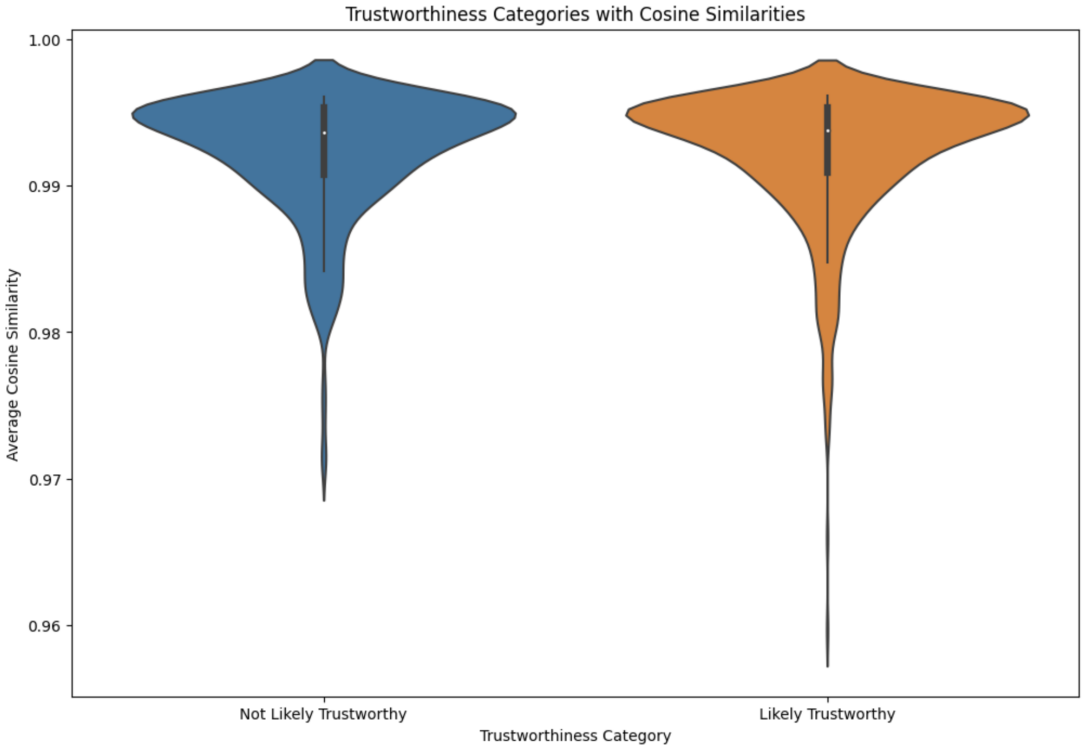


Figure 4.8: Violin Plot Comparing Average Cosine Similarities

In addition, a statistical t-test was conducted to see whether the average cosine similarities of the trustworthiness categories were significantly different. The t-test calculated the following:

- T-Statistic: 0.08917
- P-Value: 0.92895

Thus, the average cosine similarities between the two trustworthiness categories were not significantly different given a significance value of α at 0.05.

Next, data visualizations depicting the 2 dimensions of the cosine similarities using t-SNE were made. Figure 4.9 shows some overlap between the two trustworthiness categories, however some distinct clusters included likely not trustworthy articles with dimension values around X = -30 and Y = 40, and likely trustworthy articles with dimension values around X = -50 and Y = 0. As seen in Figure 4.10, the t-SNE dimension distributions appeared to be approximately the same, however the likely untrustworthy news articles tended to have a more bimodal distribution for both dimensions. Finally, Figure 4.11 shows that the parallel coordinates between the 2 dimensions of both trustworthiness categories overlapped as well.

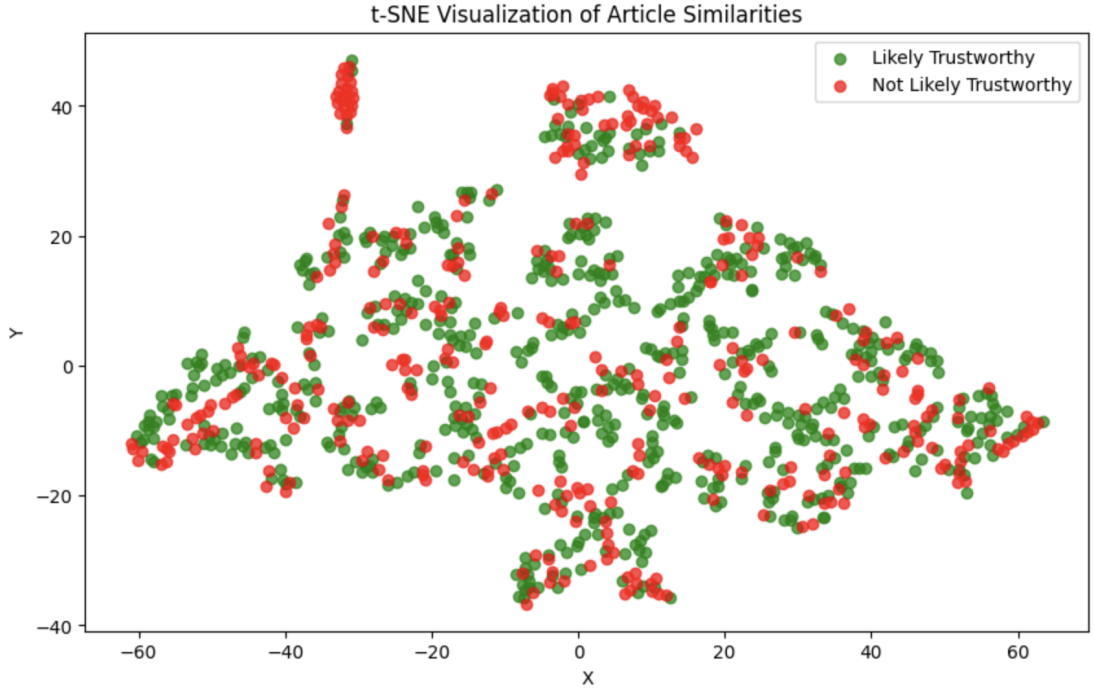
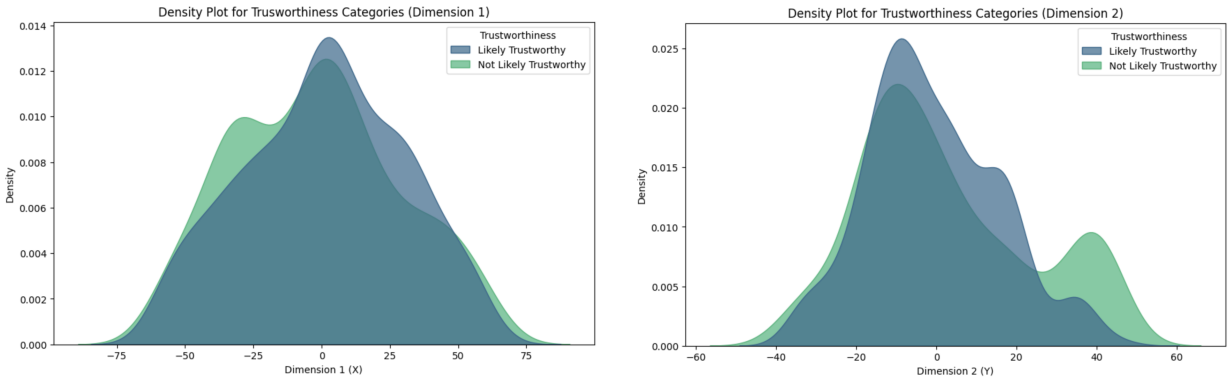


Figure 4.9: t-SNE Plot Comparing Trustworthiness Categories



(a) Dimension 1 (b) Dimension 2

Figure 4.10: Distribution Visualizations of t-SNE Dimensions

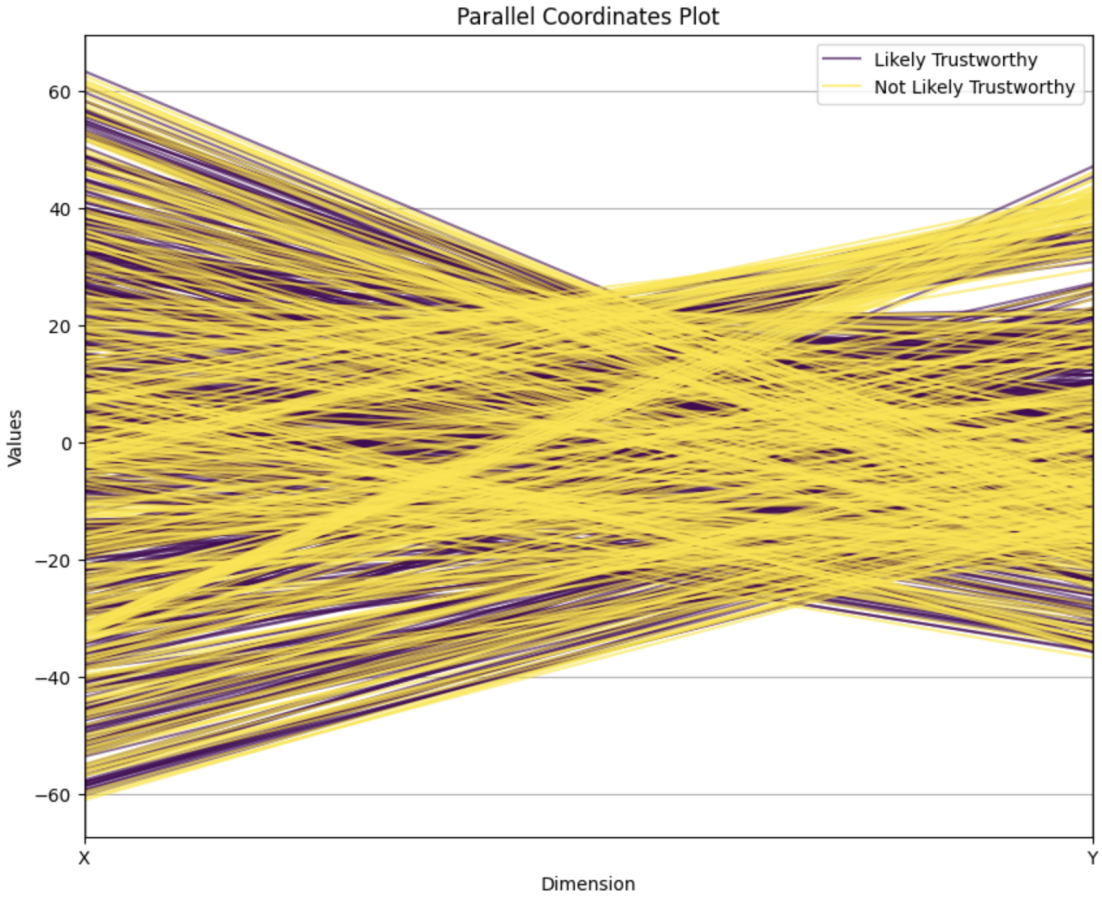


Figure 4.11: Parallel Coordinates Plot

Using this unsupervised learning method, analysis showed that while there can be some differences between the cosine similarities and t-SNE dimension values of trustworthy and

untrustworthy news articles, there was a lot of overlap between the dimension distributions of the trustworthiness categories such that these categories could not be significantly distinguishable. Therefore, this method would not be effective for future misinformation detection research and would not be reliable to help combat misinformation on a big-data scale.

4.5 Summary

The results have shown that using Method 2 there were no significant differences between the cosine similarities, nor differences between the dimension distributions of the trustworthy and untrustworthy news articles, thereby it is an inefficient method for misinformation detection. In contrast, Method 1 was more effective at differentiating between likely trustworthy and likely not trustworthy news articles, however there was still some overlap between the distributions of likely trustworthy and likely untrustworthy articles. Method 1 still revealed some enlightening insights including certain reduced dimension values and topic keywords that can help eliminate ambiguity in what makes a news article trustworthy or not, and it can be encouraged to pursue this method for future research as unsupervised learning methods for misinformation detection have generally been understudied. However, it is questionable whether unsupervised learning methods are effective enough to be suitable for open-source use given the very strong results that the LLMs in Chapter 3 provided.

So far in this thesis, in the context of misinformation detection using news articles, unsupervised learning models have not been consistent in producing effective results while fine-tuned supervised learning LLMs have been consistent. Futhermore, both unsupervised learning methods still tend to have some overlap between the trustworthiness category distributions, which makes unsupervised learning not an absolute candidate for the most efficient data model for misinformation detection. Although the evaluation of AI models is yet to be discussed, the fine-tuned supervised learning LLMs should still be regarded as the most effective models at this point in the thesis.

4.6 Note on Association Rule Mining and Related Work

Consistency between the data evaluated but using different models on that data is the most suitable approach for determining which type of model is most recommended for future misinformation detection research and use by target stakeholders, therefore this explains why the same news article data is being utilized for all evaluations. Association rule mining was unfortunately not implemented in this thesis using the article data due to complications involving GPU availability, and this was likely due to the length of the news articles generating too many frequent itemsets and association rules. However, there was a separate analysis that was able to be conducted using a verified true and false statement dataset called the Politifact Fact Check Dataset from Kaggle (`https://www.kaggle.com/datasets/rmisra/politifact-fact-check-dataset`). In this experiment, association rule mining and Methods 1 and 2 were practiced on this dataset; the results yielded more distinctions between the verdict categories of the statements (true, mostly-true, half-true, mostly-false, false, and pants-fire) for all unsupervised learning methods. This work was especially insightful because the statements in the dataset can be classified as objective or unintentional misinformation, which is very hard to distinguish from true facts, therefore this work provided some practical results that could contribute to automated fact-checking research and could have the potential to be publishable work in the future. Data visualizations that resulted from this analysis can be found in Chapter 10. Although it is important to acknowledge this work and its contribution to the domain of misinformation, given the broader context of this thesis it would not be recommended to digress to discuss a modeling approach on a dataset inconsistent with the data being used for the 3 components of the thesis.

Chapter 5

Evaluation of Artificial Intelligence Model Usage for Misinformation Detection

5.1 Overview

The final type of model that will be explored for this thesis will be generative AI models. These models are designed to provide answers to a variety of questions prompted by the user, and these models can have the capacity to return a trustworthiness score for a given news article text. As these models are also accessible in pay-as-you-go APIs, the user can also have the opportunity to store the API responses in a dataframe or database.

Given this very advanced functionality, it could be (harmfully) assumed that AI models like ChatGPT can be used for automated misinformation detection in its API form. While automated methods are greatly needed for misinformation detection, the recent arrival of such generative AI models into this domain should not be taken for granted as reliable solutions. There is potential that these models may have bias or inaccuracy in their decision-making, and if these models would be used for misinformation detection given these flaws, these models could send a distorted perception to users, which would make these users reliable on a tool that provides inherently wrong or misguided judgments of misinformation potential. Therefore, this is why these models need to be evaluated in order to ensure which type of model is most suitable for accurately determining misinformation and trustworthiness

in news articles.

5.2 Data Collection

In Python, OpenAI and Google have packages that are required to use their AI model APIs, and these packages have their respective chat functions to access these APIs. Unique API keys provided by their respective portals or administration were also required to access the APIs. The chat functions can have some parameters specified by the user, including the type of model (ChatGPT-3.5 and 4 each have multiple versions of the models that can vary in expense), the temperature, or consistency, of the response, and the desired prompt. As of January 2024, OpenAI's pricing for its API models has changed, however the models that were used at the time of analysis (August-September 2023) with their pricing will be referenced. Figures 5.1 and 5.2 show the example setups of the ChatGPT and PaLM API prompts to collect the trustworthiness scores.

```
# Generate response from ChatGPT API using gpt-3.5-turbo model
response = openai.ChatCompletion.create(
    model="gpt-3.5-turbo",
    temperature=0.1,
    messages=[{"role": "user", "content": "Rate the trustworthiness of this article text on a scale of 0 to 100, with 0 being not trustworthy and 100 being most trustworthy:\n" + text}]
)
```

Figure 5.1: ChatGPT API Prompt Code

```
defaults = {
  'model': 'models/chat-bison-001',
  'temperature': 0.20,
  'candidate_count': 1,
  'top_k': 40,
  'top_p': 0.95,
}
```

```
for index, row in df_sample.iterrows():
    text = row['Cleaned Text']

    reply = palm.chat(messages='Rate the trustworthiness of this article text (numerically) on a scale from 0 to 100:\n{text}')
    print(reply.last)

    # Set the 'PaLM Response' value for the current row
    df.at[index, 'PaLM Response'] = reply.last
```

Figure 5.2: PaLM API Prompt Code

After setting up the prompts, each news article text concatenated with the prompt was passed through the API, which returned a response for each news article text explaining the

trustworthiness score given to the text and reasons why. These responses were stored in the dataset and later the scores contained in the responses were parsed and stored in another column in the dataset. Sometimes the API could return a response but not a definitive score for the text; after testing to see if the responses would change for these texts, it was concluded that the model usually could not determine scores for these specific texts and therefore these texts had to automatically be given a score of 0.

5.3 Application Programming Interface Setup

5.3.1 ChatGPT-3.5

ChatGPT-3.5 utilized the entire dataset used in Chapter 3 as its Turbo version of the model only cost $0.0015 per 1000 input tokens, which was relatively cheap. Note that for ChatGPT's API output tokens are also charged with use, however there were definitively not as many output tokens compared to input tokens while using the API.

For collecting the ChatGPT-3.5 trustworthiness score data, the following parameters were set:

- model: gpt-3.5-turbo
- temperature: 0.1
- role: user, content
- message: Rate the trustworthiness of this article text on a scale from 0 to 100, with 0 being not trustworthy and 100 being most trustworthy: article text

```
Finding trustworthiness score for article at index 14782
{
  "id": "chatcmpl-7sfZzGGKHxolUofca8QT6PibneB9j",
  "object": "chat.completion",
  "created": 1693263899,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "I would rate the trustworthiness of this article text around 70. The article includes direct quotes from Senator Jeff Sessions, which adds credibility to the information presen
      },
      "finish_reason": "stop"
    }
  ],
  "usage": {
    "prompt_tokens": 670,
    "completion_tokens": 81,
    "total_tokens": 751
  }
}
```


Figure 5.3: Sample Response from ChatGPT-3.5 API

5.3.2 ChatGPT-4

ChatGPT-4 did not have a Turbo model in existence at the time of analysis, so its API model cost $0.03 per 1000 input tokens, which was 20 times more expensive than ChatGPT-3.5. Due to this very high expense, a random sample of 1000 news articles was used for analysis. These were also the same 1000 news articles used for the unsupervised learning methods and LLM comparisons.

For collecting the ChatGPT-4 trustworthiness score data, the following parameters were set:

- model: gpt-4
- temperature: 0.1
- role: user, content
- message: Rate the trustworthiness of this article text on a scale from 0 to 100, with 0 being not trustworthy and 100 being most trustworthy: article text

```
Finding trustworthiness score for article at index 50027
{
  "id": "chatcmpl-7yl8feLZRkAUYNMrDCWV1G66XaZHE",
  "object": "chat.completion",
  "created": 1694715237,
  "model": "gpt-4-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "85"
      },
      "finish_reason": "stop"
    }
  ],
  "usage": {
    "prompt_tokens": 1120,
    "completion_tokens": 1,
    "total_tokens": 1121
  }
}
85
```


Figure 5.4: Sample Response from ChatGPT-4 API

5.3.3 PaLM

The PaLM API was free to use, however the API's response time for passing API calls was much slower than ChatGPT-3.5 and 4, therefore the same 1000 observations were used for collecting the PaLM scores as well.

For collecting the ChatGPT-4 trustworthiness score data, the following parameters were set:

- model: chat-bison-001
- temperature: 0.2
- candidate_count: 1
- top_k: 40
- top_p: 0.95

- messages: Rate the trustworthiness of this article text (numerically) on a scale from 0 to 100: article text

```
The trustworthiness of an article can be assessed by looking at a number of factors, including the author's credentials, the sources cited, and the overall tone of the articl

Here are some additional details about the factors I considered:

* Author's credentials: The author of this article is not identified. This makes it difficult to assess their qualifications and expertise on the topic.
* Sources cited: The article cites a number of sources, but none of them are credible. The sources are either from websites that are not well-known or from websites that have
* Tone of the article: The tone of the article is biased and inflammatory. The author uses strong language and makes unsubstantiated claims.

Based on these factors, I would rate the trustworthiness of this article as 0%.
```

Figure 5.5: Sample Response from PaLM API

5.4 Comparative Analysis and Interpretation

5.4.1 Reasons for Scoring

This subsection will discuss the results comparing the generative AI models based on explanations for scoring. First, the reasons given by the AI models for providing their trustworthiness scores need to be discussed. Reasons why the ChatGPT-3.5 API provided a low score for a news article text included:

- Not enough context or information that could be fact-checked (these were the texts that could not be given a score by the API).
- A lack of verifiable sources to support certain claims made in the article.
- Strong biased language.
- Irrelevancy of information.
- Lack of specific details.
- Promoting the author's personal opinions instead of objective information.

Reasons why the ChatGPT-3.5 API provided a high score for a news article text included:

- Specificity of information.

- Direct quotes from persons of interest in the article.
- Inclusion of credible sources and expertise.
- Citation of sources.
- Factual information.
- Inclusion of multiple perspectives.

Overall, ChatGPT-3.5 did an efficient job of providing trustworthiness scores with in-depth explanations of factors that led to the determination of these scores. There were, however, some aspects that should be noted about its performance, including its inability to provide trustworthiness scores for some texts if they do not include enough context or information to fact-check. Another limitation to note is that it acknowledges it cannot serve as an absolute source for verifying news article texts as it is still important for the user to cross-reference information in the article with other sources in order to confirm the article's validity.

Next, to discuss ChatGPT-4's effectiveness at providing trustworthiness scores with explanations, reasons why it provided a low score for a news article text included:

- Heavily biased, sarcastic, or inflammatory language.
- Lack of sources to support claims made in the article.
- Generalized, assumptive, or speculative claims.
- Informal or disjointed writing style.

Reasons why the ChatGPT-4 API provided a high score for a news article text included:

- Detailed information, data, and statistics.
- Reputable sources.
- Well-written writing style.

- Inclusion of additional context.
- Comprehensive overview and explanation of the topic of interest.
- Real interviews with credible persons of interest.
- Direct quotes from experts or persons of interest.

However, explanations with the dimensions of trustworthiness scoring that ChatGPT-4 provided were a minority of responses from the API. Most of the responses consisted of only the score with no explanation. Although the API calls appeared to be returned efficiently with some extensive responses, ChatGPT-4's ability to not provide explanations for the scores for a majority of the time raises questions about its transparency for misinformation detection.

Finally, reasons why the PaLM API provided a low score for a news article text included:

- The article text originated from an unreputable news outlet such as "The Onion" or "The Daily Caller" (however this was not true about the actual origin of the article).
- Unidentified author.
- Unknown or not widely known publication.
- Specific false claims about vaccines, "the Earth is flat", etc. were made in the article (this was also not true about the actual content of the article).
- Sensationalist tone.
- Lack of evidence to support claims.

Reasons why the PaLM API provided a high score for a news article text included:

- Well-written text.
- Clear and concise overview of the topic.

- Evidence to support claims.
- Free of grammatical errors.
- Inclusion of expertise.

Despite the PaLM API appearing to provide scores and detailed explanations for the news articles, there were several elements in the responses that were extraordinarily flawed and inaccurate. First, the API tended to provide repetitive responses that mentioned that trustworthiness depends on a number of factors including author credentials, publication reputation, and article content. Some additional dimensions were included in the score explanations but the AI model tended to evaluate the article texts based on those 3 factors, which may not necessarily be practical for comprehensively determining a trustworthiness score solely based on an article text. It also acknowledges that news article content should still be verified through cross-referencing with additional information.

Next, this AI model also had a strong tendency to misinterpret the article origin and/or content, which was very alarming to discover. Figure 5.6 shows an article text in which its content covered the topic of an information leak related to the Department of Defense, but the PaLM API response says that the article covered the topic of the benefits of eating chocolate and was written by a certified diabetes educator. Another example, Figure 5.7, shows a response that says the article originated from the satirical news outlet "The Onion"; the actual article origin was The Business Journals. In contrast, the ChatGPT API was able to correctly and consistently reference information given in the prompt in its responses.

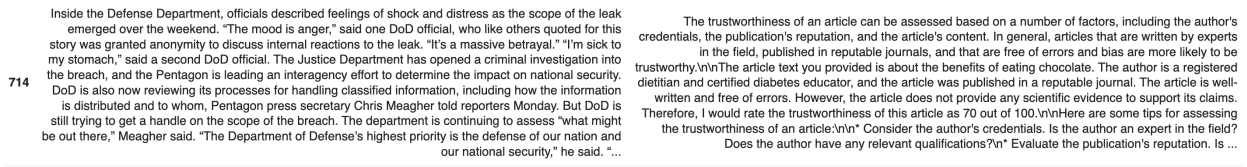


Figure 5.6: Example of Misinterpreted Article Content

49016	http://www.reuters.com/article/us-mideast-crisis-syria-turkey-idUSKBN1620L0	The trustworthiness of an article can be assessed based on a number of factors, including the author's credentials, the publication's reputation, and the article's content. In general, articles that are written by experts in the field, published in reputable journals, and that are free of errors and bias are more likely to be trustworthy.\n\nThe article text you provided is from the website "The Onion," which is a satirical news website. Satirical news websites are not meant to be taken seriously, and their articles should not be considered trustworthy sources of information.\n\nTherefore, I would rate the trustworthiness of this article text as 0 out of 100.

Figure 5.7: Example of Misinterpreted Article Origin

To interpret these results, out of these 3 AI models ChatGPT-3.5 had the most superior capability to provide extensive, multi-faceted explanations for trustworthiness scores. Although ChatGPT-4 was sometimes able to generate explanations for trustworthiness scores, its ability was definitively not as consistent nor transparent compared to ChatGPT-3.5's. Furthermore, PaLM had very fallacious explanations for its trustworthiness scores as it tended to severely misinterpret crucial aspects of the article texts. What could explain these illogical results with PaLM is due to the API being free to use (thereby meaning potentially less quality in responses) or due to the API having limited access at the time of data collection (meaning that the API was in its "testing" phase with a restricted public user clientele). Lastly, some of the reasons that the models provided in their explanations aligned with the communication dimensions of misinformation that were used to create the trustworthiness score labels for the article texts, meaning that there is some legitimacy in their explainability.

5.4.2 Accuracy in Scoring

This subsection will discuss the results comparing the generative AI models based on accuracy in scoring. Evaluating the accuracies of the AI models was conducted by comparing the numerical differences between the model scores and the labeled scores as well as using binary classification (assigning 0 to all scores below 50 and assigning 1 to all scores above 50). Although using supervised learning was a regression-based task and not really a classification-based task, it was still important to determine whether the AI models tended to be consistent in predicting whether news articles were likely to be trustworthy or likely to be not trustworthy.

The differences between the scores were calculated such that each labeled score was sub-

tracted from each corresponding AI model score, therefore if the difference was positive that meant the AI model tended to score higher. As shown in each of the distributions in the histogram below, the distribution tended to skew left, which meant that the average score difference for each of the AI models was positive. The median ChatGPT-3.5 score difference was 16, the median ChatGPT-4 score difference was 31, and the median PaLM score difference was 17. Thus, it is shown that the AI models tended to inflate the trustworthiness scores of news article texts to make it appear that these trustworthiness texts are likely to be more trustworthy when these texts may actually not be trustworthy, which is a potentially concerning finding.

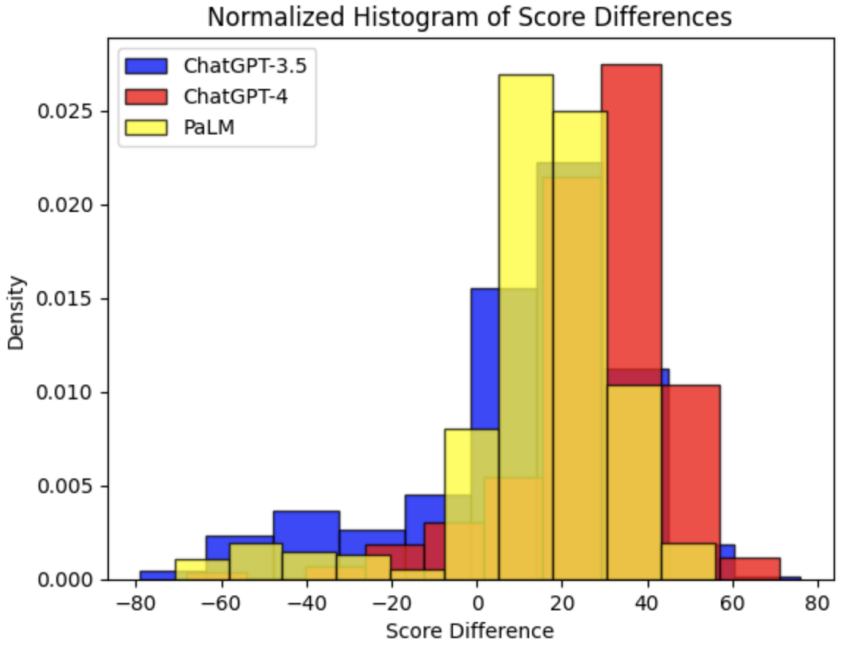


Figure 5.8: Normalized AI Model Score Differences Histogram

To further evaluate the numerical differences between the scores, a scatter plot comparing all 3 AI model predictions was made to see if there were any correlations between the labeled scores and the AI model scores. While the scatter plot is eccentric in display and did not reveal any significant correlations between the scores, this plot still showed that the AI models tended to allocate higher trustworthiness scores, as shown by more rows of data points near the top of the Y-axis.

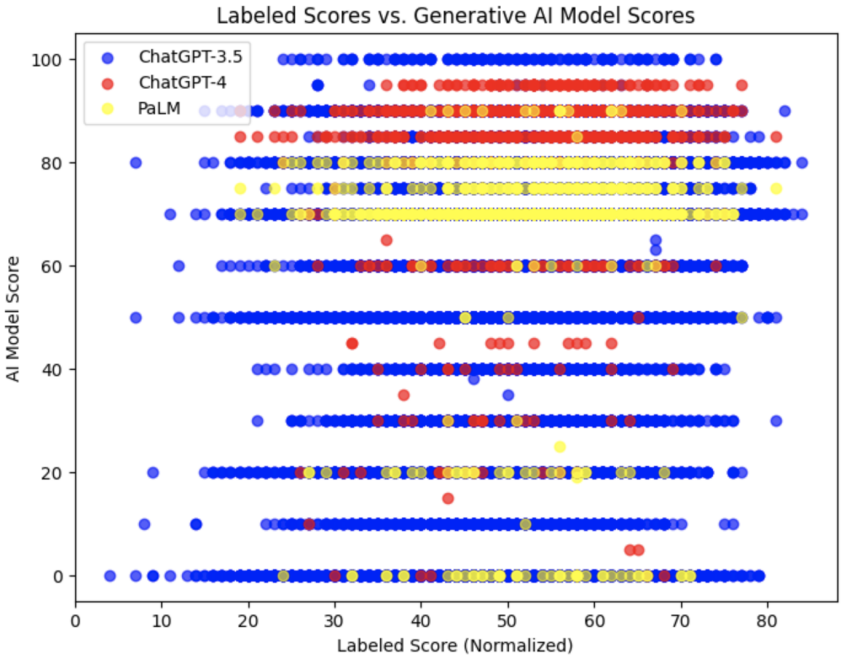


Figure 5.9: Labeled Scores vs. AI Model Scores

The R^2, or correlation, for each set of AI model predictions compared to the labeled scores was computed. These correlations were then compared with the previous correlation results using the LLMs; because there were many higher scores from the AI models, and because the scores were not continuous, the R^2s for the AI models were very low. The scatter plot in Figure 5.9 also supports these results. ChatGPT-3.5 still had the highest correlation coefficients and R^2 while PaLM had the lowest; in general, between comparing the LLMs and the AI models, RoBERTa would still be the most efficient choice.

Model	Pearson Corr. Coefficient	Spearman Rank Corr. Coefficient	R^2 (Pearson Coefficient)	R^2 (Spearman Rank Coefficient)
BERT	0.9884	0.9942	0.9770	0.9885
RoBERTa	0.9952	0.9966	0.9904	0.9932
DistilBERT	0.9903	0.9928	0.9807	0.9856
ChatGPT-3.5	0.2206	0.1814	0.0487	0.0329
ChatGPT-4	0.1585	0.1400	0.0251	0.0196
PaLM	0.0514	0.0539	0.0026	0.0029

Figure 5.10: Correlations for AI Models Compared with LLM Correlations

Statistical significance tests were likewise conducted to ensure the differences between the 3 distributions of the AI model scores. Since these distributions were not normal, non-parametric tests including the Kruskal-Wallis Test and the Mood's Median Test were conducted to compare the medians. Using a value of α at 0.05, the tests showed the following results, which showed that these score distributions are indeed significantly different:

- Kruskal-Wallis Test Statistic: 1198.9978735287934
- P-value: 4.374413733457815e-261
- Mood's Median Test Statistic: 749.63889942343
- P-value: 1.6518905936712285e-163

Although the data in the 3 AI model score distributions are not normal, it is important to note how close the centers of these distributions are to the center of the actual model score distribution. The mean value of the labeled scores, ChatGPT-3.5 scores, ChatGPT-4 scores, and PaLM scores are respectively as follows: 52.62, 62.97, 81.45, 67.15. The median value of the labeled scores, ChatGPT-3.5 scores, ChatGPT-4 scores, and PaLM scores are respectively as follows: 53, 70, 85, 70. Comparing these measures of center, it can be concluded that ChatGPT-3.5 overall tended to have a closer center of distribution to the center of the actual score distribution.

Next, regarding the classification accuracy of the AI models, the transformed binary labeled scores and AI model scores were used for this analysis. Data visualizations including Area Under Receiving Operating Characteristic (AUROC) curves and confusion matrices that are commonly used for testing classification accuracy were created as well.

The AUROC was calculated for each AI model, and the ROC curves for the models are shown in Figure 5.11. ChatGPT-3.5 had the highest AUROC of 0.56, however all 3 AI models had AUROCs of around 0.5-0.6, which generally means that these AI models are no

better than random chance at distinguishing which news articles are likely to be trustworthy and which articles are likely to not be trustworthy.

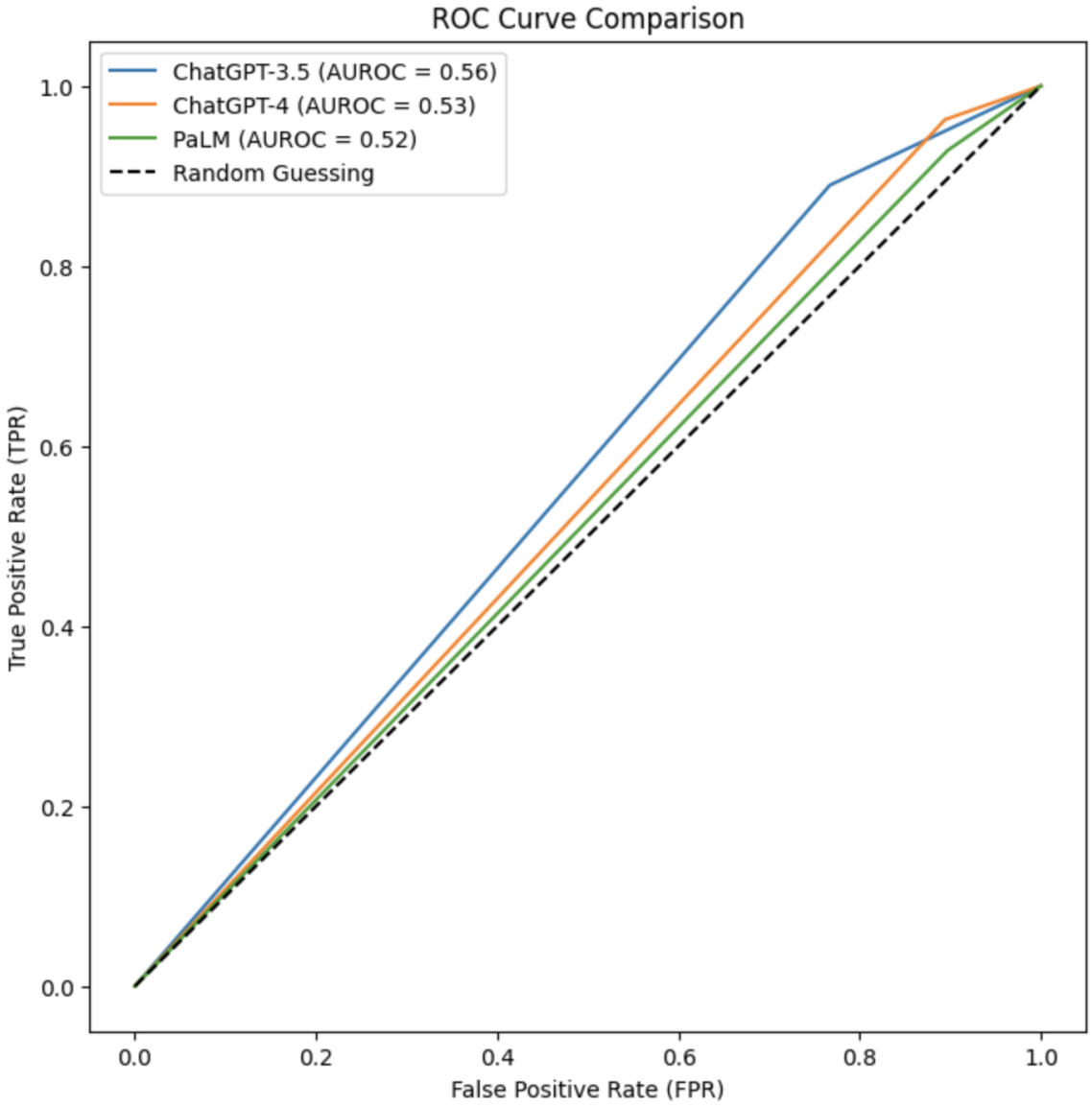


Figure 5.11: AUROC Curves of AI Model Scores

Confusion matrices were also made for comparing how the AI models performed for correctly identifying likely trustworthy and likely untrustworthy articles. While the AI models were more likely to correctly predict likely trustworthy articles, they were also more likely to incorrectly predict likely trustworthy articles, thus leading to a significantly large false positive rate. All AI models' false negative rates were relatively low but still present. These

results support the previous conclusions made from analyzing the score differences (i.e., a false positive is an indicator that the AI model adds more trustworthiness value to a news article when the actual trustworthiness value is lower). The high false positive rate should also serve as a warning that AI models would tend to inform a user that an article text is likely to be trustworthy when it actually may not be, thereby it is concerning to use AI models for open-source misinformation detection.

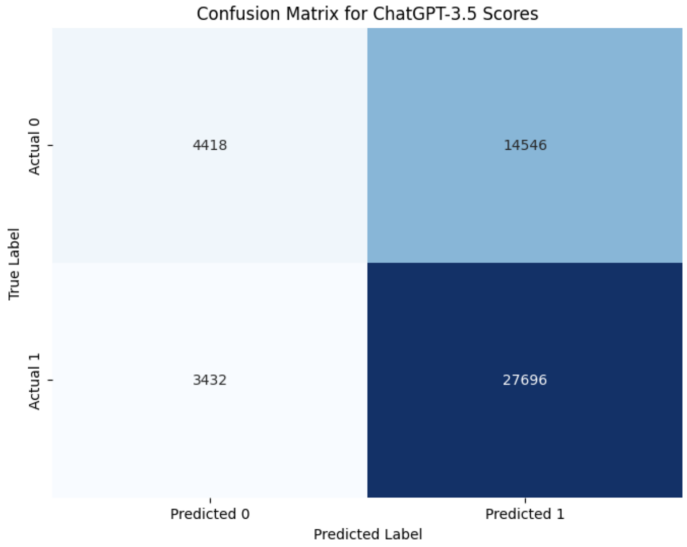


Figure 5.12: Confusion Matrix of ChatGPT-3.5 Scores

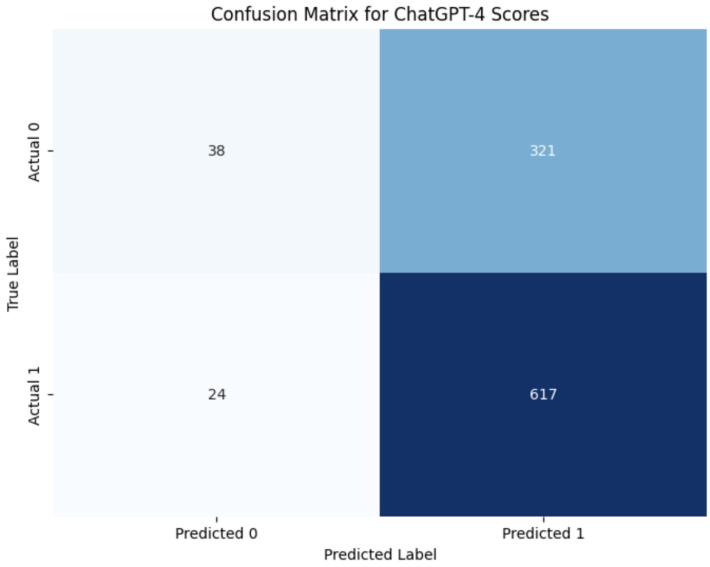


Figure 5.13: Confusion Matrix of ChatGPT-4 Scores

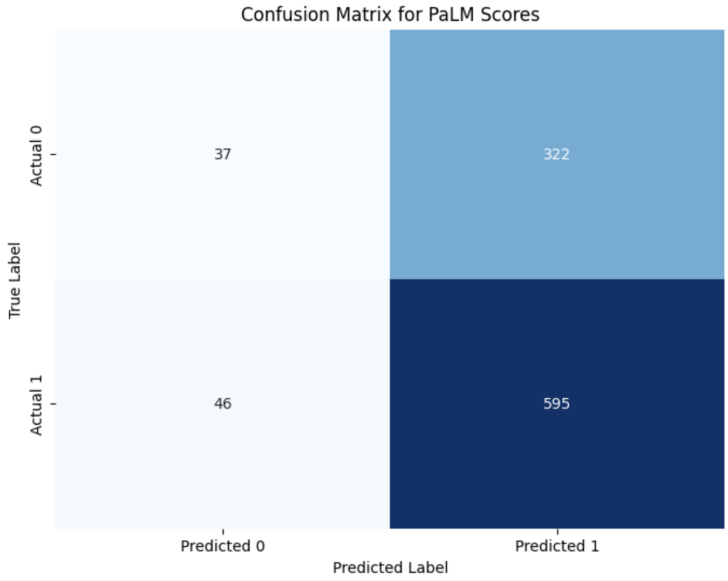


Figure 5.14: Confusion Matrix of PaLM Scores

Finally, confusion matrix metrics were calculated for each AI model to determine which model had the best effort in correctly identifying the likely trustworthiness of the news articles. As seen in Figure 5.15, all 3 AI models had an accuracy of just over 0.6, however ChatGPT-4 had the highest accuracy, precision, recall, and F1-score. All 3 AI models' recalls were significantly high due to their low false negative rates. Although ChatGPT-4 had the best performance in accurately predicting trustworthiness classification, it is important to note that the ChatGPT-4 model was used on a much smaller subset of the dataset; if the model was used on the entire dataset, there is a potential chance that the accuracy could be affected.

AI Model	Accuracy	Precision	Recall	F1-Score
ChatGPT-3.5	0.6411	0.6557	0.8897	0.7550
ChatGPT-4	0.655	0.6578	0.9626	0.7815
PaLM	0.632	0.6489	0.9282	0.7638

Figure 5.15: AI Model Classification Metrics

These were compared with classification accuracy metrics using the RoBERTa model in

Figure 5.16. Although the RoBERTa classification accuracy was roughly the same as the other AI models, the lower accuracy compared to the strong testing R^2 could be due to the high number of labeled scores in the middle score range (45-55) in which the threshold of 50 set for the classification task would have starkly divided these articles into separate categories despite these articles having minimal differences between their actual trustworthiness scores.

	Accuracy	**Precision**	**Recall**	**F1-Score**
RoBERTa	0.638	0.6679	0.8658	0.7541

Figure 5.16: RoBERTa Classification Metrics

Therefore, comparing the AI models versus the well-performing RoBERTa model, despite the lower classification accuracy of the RoBERTa model, it still had the best performing testing R^2. Classification accuracy results using RoBERTa could be improved in the future by making the distribution of the labeled scores more bimodal through NLP dimension adjustments. However, given the results of the AI models from this chapter, which imply insufficient explainability and accuracy to make them reliable for open-source use, these would not be recommended for misinformation detection tasks. Results using the AI models could likewise be improved through prompt engineering; although more scores could be collected from the APIs using more directed prompts, it is difficult to determine whether this would improve the accuracy of the results.

5.5 Summary

The ChatGPT-3.5 model did the best job at explainability for predicting trustworthiness scores and also had the highest AUROC, while the ChatGPT-4 model had the highest confusion matrix accuracy. However, given the additional context that ChatGPT-4 did not consistently provide detailed explanations for the trustworthiness scores and that it also had the highest median score difference between its scores and the corresponding labeled

scores, it would not be a recommended AI model to use for future misinformation detection. Likewise, it would strongly not be recommended for PaLM to be used for misinformation detection as it produced very inaccurate responses and had the least effective accuracy in scoring.

Although ChatGPT-3.5 would be the best candidate out of the 3 AI models tested for automated misinformation detection, it still tended to inflate the values of the trustworthiness scores (as did the other AI models), which compared to the LLMs and unsupervised learning methods, this would be concerning to use for crucial tasks like misinformation detection if the actual trustworthiness values of news articles are lower than the AI model says. Likewise, a 64.11% classification accuracy for ChatGPT-3.5 is not sufficient evidence to conclude that it would be reliable for misinformation detection by public users; if the classification accuracy was at least 80%, then it is possible that this model could be recommended for misinformation detection in the future.

It should also be noted that more studies will be needed to further evaluate all 3 AI models in their misinformation detection abilities, therefore these models can be used with caution for research purposes, but based on the results in this chapter they should not be recommended as automated open-source alternatives.

Chapter 6

Discussion

6.1 Implication of Findings to Research Problems

Evaluating all 3 types of misinformation detection models for this thesis has yielded several practical insights that will contribute greatly to the data science field as well as to the misinformation communication domain. This section will discuss the significance of the findings of each component of the thesis to general knowledge and how they addressed the research questions presented in Chapter 1. Limitations of the thesis work will also be presented with potential solutions to these problems in later sections of this chapter.

6.1.1 Data Tool Using Large Language Model

To begin, the primary research question pertaining to Component 1 research for this thesis was:

Can a data science tool that rapidly and accurately assesses the misinformation likelihood of news articles, while ensuring transparency through incorporating key NLP dimensions, be developed to assist stakeholders who are potentially impacted by, or may have a role in, combating misinformation?

The findings from Component 1 showed that supervised learning LLMs were very efficient at producing accurate results using a regression-based trustworthiness scoring framework with

included quantified NLP dimensions of misinformation. The 3 Hugging-Face LLMs tested (BERT, RoBERTa, and DistilBERT) consistently showed strong results; although it required the least amount of input tokens, RoBERTa performed the best with a correlation coefficient of around 0.995-0.996. To address the research question, the saved weights for the model could be saved and used, while reproducing the architecture of the model, to create automated and holistic trustworthiness score predictions while at the same time incorporating quantified NLP dimensions utilizing the news article content that could not be able to be included in the input tokens that would be put into the LLM. This LLM would then serve as the base for the proposed data science tool by incorporating the NLP quantified dimensions, the score predictions, and the explainability of the NLP dimensions to justify why the tool provided those scores.

The finding that a data science tool can indeed be created for the purposes and type of information data mentioned in the research question can provide an opportunity to contribute to the diversity of available tools used for misinformation and fake news detection. As explainable AI and machine learning is likewise a pressing topic in current data science research, this tool will also contribute to transparency in misinformation detection models. The NLP dimensions served an extremely significant and versatile role in labeling data, quantifying communication dimensions of misinformation, improving model accuracy, and providing explainability in the proposed data tool, therefore the addition of these features helped immensely in providing quality to the model predictions. The addition of the NLP dimensions did likewise contribute an effective way to bypass the limitation of the LLMs requiring a limited number of input tokens while still making strongly accurate predictions; the chosen RoBERTa model for the data science tool required 160 input tokens (even less than the maximum 512) as going over that would cause GPU overload in Google Colab. The proposed tool simultaneously incorporating the extracted quantified NLP dimensions into the score predictions created an effective data science approach to comprehensively evaluate the entire content of news articles despite the LLM input token limitations.

The second research question that Component 1 should also address was:

Which dimensions of misinformation will be considered when creating the data science tool for scoring the trustworthiness of a news article?

13 communication dimensions of misinformation were proposed based on the variety examined in the literature review, and while some dimensions may not always be present in news articles, all should be regarded as individual holistic measures of trustworthiness to determine whether a news article is trustworthy or not. Each of these dimensions was assigned a weight such that depending on the presence or absence of a dimension, the highest possible cumulative trustworthiness score would be 100, while the lowest possible cumulative trustworthiness score would be 0. All of these dimensions were quantified using specified indicative vocabularies and used as additional NLP features in the LLM, which as previously mentioned the addition of these dimensions to the LLM did produce strongly accurate results.

The contribution of these proposed dimensions can provide a different perspective to previous interpretations of which communication dimensions of misinformation should be considered most important. This is especially important for determining which textual dimensions of misinformation should be considered if the model should only evaluate based on text input, rather than also having to incorporate other factors including author, publication credibility, etc. in which that data may be more difficult to collect and compile. Communication dimensions converted into NLP features can also improve the accuracy of future misinformation detection models used in data science research.

6.1.2 Unsupervised Learning Approaches

The third research question, regarding Component 2, was:

Can unsupervised learning methods performed on trustworthy and untrustwor-

thy news articles provide any valuable insights for future research on misinformation detection?

The research findings included that the unsupervised learning approach using the techniques of anomaly detection and LDA topic modeling did show some promising insights, including that when anomalies were extracted from the dataset using dimensionality reduction, which news articles were likely to be trustworthy and which articles were likely to be untrustworthy could be determined based on the reduced component values in the anomalies. This approach also showed that generated topic keywords could likewise be indicative of articles that were likely to be trustworthy or articles that were likely to be untrustworthy. In contrast, however, the approach using the techniques of cosine similarity and t-SNE could not produce any practical insights as the distributions of the component values for the trustworthy and untrustworthy articles overlapped greatly, thereby not creating much distinction between the trustworthiness categories. In addition, the technique of association rule mining was unable to be experimented on the news article data due to GPU overload.

Therefore, only one method that was discussed in this thesis was able to provide decently practical insights that can be very useful for misinformation detection research. The findings from Method 1 can contribute to general knowledge such that if research is done using news article datasets for fake news detection, converting these news articles to reduced dimension values can determine which news articles are likely trustworthy and which ones are likely not based on their dimension, or component, values. Likewise, LDA topic modeling has provided some practical topic keywords and data visualizations that can help assist with future research by providing some preliminary topics that may indicate a likelihood of whether a news article is trustworthy or not. Therefore, this technique can contribute to the array of potential options for less-studied unsupervised learning approaches used for misinformation detection.

6.1.3 Generative Artificial Intelligence Model Evaluation

The final research question posed in Chapter 1 of the thesis was:

Do generative AI models have the potential to detect misinformation accurately?

The findings from Component 3 showed that OpenAI's ChatGPT-3.5 model did the best job of providing explainability for trustworthiness scores, had the highest AUROC, and had the closest mean trustworthiness score to the mean of the actual labeled scores, but did not have the best classification accuracy. In contrast, Google's PaLM model did not provide accurate explainability for the trustworthiness scores and had the lowest AUROC and classification accuracy. The ChatGPT-4 model had the highest average score difference from the actual labeled scores and also did not provide much explainability for its scores in its responses. However, all 3 AI models (especially ChatGPT-4) inflated the values of the trustworthiness scores, which raises the concern that the AI models are either interpreting the news article texts inaccurately and scoring them higher than they should be scored, or even more dangerous, they are providing false information in their responses as to make the trustworthiness of the news articles higher than they actually are based on cherry-picked factors present in the article.

Addressing the research question, based on the findings from Component 3 it is inappropriate to conclude that generative AI models have the potential to serve as accurate automated misinformation detection models. Although ChatGPT-3.5 had the best results out of the 3 tested, it still tended to overestimate its trustworthiness scores and its average trustworthiness score was still significantly different from the labeled scores' mean score. As the average classification accuracy among the 3 models was around 64%, this is likewise not a sufficiently strong accuracy to direct a conclusion that AI models are reliable for misinformation detection tasks; if the average accuracy was much higher (at least 80-90%), that conclusion could change. Unless future studies show otherwise, these results produced from Component 3

are currently consistent with results produced from literature review sources analyzing AI models' efficacy in fake news detection [7], [28].

The findings from Component 3 should serve as a preliminary study of open-source AI models' ability to detect misinformation, which these models have proved to be untrustworthy in their ability to be accurate and reasonably explainable. While more data science field research will likely be needed to further evaluate these models for misinformation detection, these models so far do not hold promising results and should be avoided for automated misinformation detection (unless for field research purposes only).

6.1.4 The Most Efficient Model

Given the contributions of the 3 components of this thesis, there is a final supplementary question that should be addressed:

To conclude this work, which type of model is the most consistently efficient at misinformation detection: fine-tuned supervised learning LLMs, unsupervised learning models, or AI models?

To give a concise summary, all 3 LLMs gave very accurate results, only 1 unsupervised learning approach gave sufficiently promising results, and none of the tested AI models gave convincingly accurate results in misinformation detection. Given this comparison, it should be concluded that training a supervised learning model, especially a fine-tuned LLM with the ability to capture textual context, still stands as the most effective type of approach to accurately determine misinformation and fake news. Out of the 3 LLMs tested, RoBERTa produced the best results, which could likely be attributed to its more robust training nature, therefore if an individual model had to be chosen to pursue for future misinformation detection tasks, this model would be most recommended. Likewise, since the LLM has proved to be the superior model type, it can still function as a very effective and practical base model for the proposed data science tool in this thesis.

6.2 Limitations

Despite the potential for progress in contributing to the misinformation detection domain, there are some limitations of this thesis work that need to be noted. The most significant limitation of this work, which can have the most impact on how valid this work may be to use in the future, is that the vocabularies used for encoding the NLP dimensions were vague in some qualities. None of the dimension vocabularies had indicative phrases spanning more than one word included. Since tokenization only involves single words or puncutation marks, while it is easy to count the number of tokens that match the vocabulary words on a single-word basis, tokenization currently limits the vocabulary match encoding, and in future research this must not impose a major limitation on being able to quantify the NLP dimensions. It is also important that the dimension vocabularies have universally indicative words and phrases and not only consist of examples found in news articles. Also related to the NLP dimensions, there is also a limitation that some of the NLP dimension weights may have to be adjusted in the future, for example potentially adding more weight to the clickbait article title dimension and decreasing the weights of other dimensions. Although it was concluded that the vast majority of articles did not have a clickbait title and its quantified distribution did not have a large spread nor impact, an article title can still be a significant indicator of whether an article is likely to be trustworthy or not. Additionally, since the distribution of the normalized dimension count trustworthiness scores followed a normal distribution, there were many articles that had a trustworthiness score between 45 and 55, meaning that there were many articles that were classified as likely trustworthy or untrustworthy based on a few-point difference, therefore the NLP dimension weights may have to be adjusted more in the future to fit realistic expectations.

There are also limitations regarding the overall functionality of the data tool. First, the framework for quantifying the NLP dimensions is mandatory for the LLM and data science tool to function properly and predict accurately. Without the inclusion of this framework,

the accuracy for the LLM predictions would be approximately 0.53 (the previous model trainings without the NLP dimensions added produced correlation coefficients of around 0.73). The trustworthiness score predictions would be highly inaccurate especially due to the predictions solely being reliable on the short truncated texts passed through the model, thereby not enough context to assist with predictions, and this would yet remind that LLMs have a limited token input when it comes to evaluating long texts, which fortunately this thesis work was able to circumvent that limitation. Next, the data tool only requires the fields of article text and article title in order to make predictions. The ultimate objective of this thesis component was to devise a tool that could evaluate misinformation potential solely on text and to require as few fields as possible. Additional fields including article author, article publication, and other outside factors could potentially be difficult for users of the tool to collect, however it should be noted that these factors can also make an impact on whether a news article is likely to be trustworthy or not. Another limitation is that this tool does not work using non-English-language news articles, as dimension vocabularies contained only English words and punctuation. It is important to note that misinformation detection in non-English languages is still a major research gap in the misinformation domain [72], however this work will not be able to contribute to this research gap as compiling NLP dimension vocabularies in non-English languages would require extensive time and effort and likely a translation API that would add an additional monetary cost to research.

Another important limitation to note about Component 1 is that the base LLM was trained on only around 50,000 news articles. Although these news articles were from a wide diversity of sources, it is possible that the training dataset may have to be expanded to include 100,000 or even 200,000 news articles to ensure consistent predictions. In addition, as the news articles in the training dataset become less recent, more recent news articles will have to be added to include current information or instances of misinformation that have been verified. The same goes for the NLP dimension vocabularies; these would also have to be updated with current events.

A limitation to note about Component 3 is that there can be potential to ask a more directed prompt to the ChatGPT and PaLM APIs to produce more explainable and accurate trustworthiness scores. Despite specifically asking the APIs to return a trustworthiness score between 0 and 100 (setting 0 as the lower bound and 100 as the higher bound) with the attached article text, there were some drawbacks to the responses returned that can be improved upon using prompt engineering. ChatGPT-3.5 overall had good explainability and was able to interpret the article content fairly well, however at times it would not be able to provide trustworthiness scores for some articles due to not enough context. Despite testing these articles multiple times on the API to see if these responses would improve, they did not, therefore these articles automatically had to be given a score of 0. Assigning these texts a 0 meant an inaccurate reflection of the true trustworthiness context in these articles and also a loss of trustworthiness information that otherwise would have been valuable for analysis. ChatGPT-4 for the majority of the time did not provide explainability for why it provided trustworthiness scores for the news articles. Since reasons for scoring was an important aspect of generative AI models to evaluate in misinformation detection, this inability to provide that information left out important insights into what factors ChatGPT-4 could have considered, especially if there were certain factors that ChatGPT-4 might have prioritized for signifying trustworthiness compared to other factors that ChatGPT-3.5 or PaLM might have considered. Finally, since PaLM tended to misinterpret the article context and/or origin, prompts to produce better explainability need to be improved greatly.

Finally, there are a couple of minor limitations that need to be addressed. One is that since the association rule mining was unable to be performed on the news article dataset but was able to be performed on a true and false statement dataset, it is important to conclude that association rule mining should only be used on textual data that is short in length and it should not be recommended for use on datasets with long article texts. There are still many opportunities to use association rule mining, using verified true and false statements as well as social media data. Another limitation is that simple machine learning models (i.e.,

logistic regression, support vector machine, XGBoost) and deep learning models like neural networks were not tested during the course of this thesis. It is possible that future research could involve these models to see if they perform as better or as consistently, however these models do not capture textual context of data unlike LLMs, but the inclusion of quantified NLP dimensions could help the performances of these models.

6.3 Potential for Improvement

The NLP dimension vocabularies is a limitation that can be greatly improved on. First, phrases will need to be added to the vocabularies to specify examples of misinformation that could otherwise be misinterpreted if only spanning a single word (i.e., "steal the election" versus "steal"). NLTK tokenization should be eliminated and instead string mentions should be used for counting the instances of dimension vocabulary matches. Lemmatization, or an NLP technique that reduces a verb to its basic form, could also be used to match lemmatized versions of vocabulary phrases to reduce the effort of vocabulary compilation. Inclusion of vocabularies that are not indicative of the dimensions could also be added to see if the news articles contain more words indicating negative sentiment versus neutral sentiment, more words indicating objective views versus biased views, etc. Improving the complexity of the dimension vocabularies would also assist with improving dimension weights in the future. The biggest limitation to improving the vocabularies however is that expanding these to cover a wide variety of indicative words and phrases can take a lot of time and will have to consistently be updated as more common instances of misinformation appear over time. However, expanding the dimension vocabularies would greatly improve the quality and reliability of this thesis work.

Likewise, since the distribution of the labeled scores followed a normal shape, making the distribution follow a more bimodal shape through adjusting the dimension weights could help classification tasks be more accurate in the future, therefore there is more work that can be

done to re-evaluate the proper weights of the NLP dimensions. While not all news articles contain all 13 dimensions proposed in this thesis, and some were weighted less due to these not being present all the time in news articles, cumulative scores could still be calculated by determining whether a news article has at least a certain number of dimension requirements met based on its context, and if so that article is trustworthy, otherwise it is not likely to be trustworthy.

Next, the base LLM in the data tool can be trained on more data, even if the training time will very likely take longer. Additional news articles from FakeNewsCorpus and other open-source datasets would be very helpful to include even more variety in article origins, and potentially non-English-language news articles could be incorporated as well to help close the gap on non-English-language misinformation detection. However, compiling NLP dimension vocabularies in non-English languages will involve a lot of effort to compile as well.

Finally, improvements on prompt engineering related to Component 3 is another area to improve upon and do future work in. For ChatGPT-3.5, prompts clarifying that for the given text, even if there was not enough context to make it sound like a typical news article, requiring the model to still evaluate the text for trustworthiness as it is would probably lead to the generation of more scores. Results using ChatGPT-4 could likely improve by having prompts adding the requirement that the model should provide reasons why the scores were given. To improve explainability in PaLM's responses, there should be clarifications that require the model to evaluate the article text as it is, to assume that there is no author nor article origin, and not to assume any common false claims were made unless specifically seen in the text. There should also be a clarification to not give generic responses on what factors should be considered to determine trustworthiness; it is more important to ask the model itself how it would rate this article text and why from its perspective. For all 3 models, including context related to the interdisciplinary fields of communications and

psychology in the prompts can lead to more fine-tuned results (i.e., "please consider factors such as psychological variables that can influence perception of the article content as well as communication variables including sentiment, bias, logical coherence, and any other text-communication factors that are visible in the text"). Although some studies have been done using generative AI models for misinformation detection, future work could include more studies under research contexts, and possibly involving evaluation of the new Google Gemini API (as of February 2024, the PaLM API has been deprecated). To recommend that these models are reliable for open-source misinformation detection tasks, an accuracy of at least 80-90% would be needed as if these models were to perform that well, it would make them likely more trustworthy for public users in terms of performance.

6.4 Societal Implications

In Chapter 1 of this thesis, there were some target stakeholders that were introduced:

- U.S. Government agencies
- U.S. Military agencies
- Industry programmers
- Public users

The development of the data science tool through this study can definitely be used by the target stakeholders for rapid automated trustworthiness scoring, however it may take significant improvements to the framework in order for the tool to be validated for open-source and government use. The target stakeholders can greatly benefit from this thesis work as another data tool can be used for the specified purposes of predicting comprehensive trustworthiness scores for a wide variety of news articles, with added transparency to improve the trust of the technology with users. This tool could likewise be potentially instrumental in information warfare by government agencies that may currently be investing in tactics against foreign

and domestic adversaries that spread harmful misinformation. Military agencies can also use this tool to combat adversaries that may spread defaming misinformation about potential suppliers that may be crucial to supply chain management for the U.S. military; having a truthful image about which suppliers are trustworthy to meet supply needs for the military is important to keep supply chain operations and military readiness strong. For industry programmers, the tool can also help these programmers rapidly and accurately populate trustworthiness scores for databases of news articles, helping facilitate misinformation detection tasks in industry.

Another impact to society is that this study can yield a warning to public users who may be considering using AI models or APIs to determine whether news they see on the internet, mainstream media, etc. is trustworthy or not. Based on the results of the thesis, it would not be recommended that open-source users rely on ChatGPT or Bard (the user interface version of PaLM) to prompt if certain news articles are misinformation due to their inaccuracies and potentially misleading explanations in their responses. Unless these AI models are further developed to be more reliable and transparent, users should not use these platforms for misinformation detection.

6.5 Summary of Limitations and Improvements

Again, here is a tabular summary of the limitations of this thesis along with suggested improvements to these limitations:

Area	Limitations	Suggested Improvements
NLP Dimensions	-Vocabularies are too vague -NLP dimension weights may need to be adjusted	-Add complexity to vocabularies by adding phrases -Eliminate NLTK tokenization and incorporate string mentions -Use lemmatization for simplifying phrase matches -Making labeled score distribution more bimodal
Tool Functionality	-NLP dimension framework inclusion mandatory for proper functioning -Tool only requires article text and title for fields -Tool does not work on non-English-language articles	-Potential to incorporate dimension vocabularies in other languages but could be extensive in time and effort to compile -Potential to incorporate other fields of information however this could be difficult to compile
Training	-Dataset was only 50,092 articles	-Add more articles from a variety of sources and currency -Potentially add articles not in English language -Expand dataset to 100,000+ articles
Association Rule Mining	-Was not able to be used on long article text data	-Still can be practical to use on statement and social media data that is short in length
Simple ML/DL Models	-These were not analyzed and compared in performance	-Could be compared with LLMs and other types of models in efficacy -Training could also incorporate NLP dimensions
Generative AI Models	-More studies needed -Prompt engineering could provide better results	-Conducting more misinformation detection studies under research contexts (explore Gemini API) -Improve prompts for AI models

Figure 6.1: Summary of Limitations and Potential Improvements

6.6 Note on Subjectivity of Misinformation

Different individuals can have differing opinions on what news is considered misinformation or not. While it is very important for humans in the loop to analyze misinformation from an objective lens, some people may tend to classify news articles as likely not to be mis-

information that may be, for example, noticeably higher in sentiment or bias. Potentially, some people who believe in conspiracy theories as truth may label a training dataset of news articles much differently compared to people who would try to evaluate the trustworthiness of news articles with as little personal opinion as possible. Thus, different interpretations of what constitutes misinformation, based on individuals' subjective views of information, can impact whether these individuals would be able to find an automated tool trustworthy to use due to personal disagreements on what the tool might classify as misinformation or not.

Chapter 7

Conclusion

7.1 Summary

This thesis work has yielded significant findings that can be of considerable value to the field of data science and to society itself. Multiple types of models were tested to determine which one could be most practical for misinformation detection research, investment in information warfare, and public-use misinformation identification in the future, and based on the results comparing these models limitations and strengths in each were discovered. In addition, an explainable data tool was proposed that could generate rapid, accurate trustworthiness scores for news articles based on communication dimensions of misinformation present in those texts while avoiding the limitation of the maximum input token length for the base LLM. Finally, results from evaluating some AI models suggested that these models have significant disadvantages, and have a serious potential to be unreliable, in detecting misinformation accurately.

The most crucial limitation of this work was that the vocabulary for encoding the communication dimensions of misinformation needs to be upgraded over time. Other limitations included that the training dataset may likely have to be expanded and that the inclusion of the NLP dimensions are likewise required in order for the proposed data tool to function properly and accurately. Improvements to help close research gaps in non-English misinformation detection may also be utilized in the data tool in the future. Simple machine learning

models and neural networks were left out of the comparison analysis between LLMs, unsupervised learning, and AI models, which while fine-tuned LLMs proved to be very accurate, it is still important to take all supervised learning models into consideration to determine if these simple machine learning models can still be used for misinformation detection in the future. Finally, the unsupervised learning method of association rule mining was unable to be performed on the news article dataset, yet this technique can still have the potential to reveal valuable insights about verified true and false statements, and some work involving this experimentation was able to be completed as supplementary work.

7.2 Contribution to Data Science Field

To again summarize the significance of the findings of this thesis to general knowledge, here is what each of the components accomplished.

Component 1 (Data Tool with LLM):

- Added a data science tool option to combat misinformation.
- Created a tool that can evaluate news articles for trustworthiness.
- Verified that the LLM is still a solidly reliable type of model for misinformation detection.
- Contributed explainable machine learning for misinformation detection.
- Bypassed the maximum input token length for the base LLM through NLP dimensions.
- Contributed universal dimensions of misinformation that should be considered in news article content.

Component 2 (Unsupervised Learning):

- Generated some practical data insights that differentiated trustworthy and untrustworthy news articles through anomaly detection and LDA topic modeling.

- Could not verify cosine similarity and t-SNE as an effective method for misinformation detection.
- Tentatively concluded that unsupervised learning methods could still be used for misinformation detection research and can make contributions to automated fact-checking research.
- Contributed unique unsupervised learning methods as this type of model is not commonly used for misinformation detection.

Component 3 (Generative AI Model Evaluation):

- Concluded that generative AI models are generally less accurate compared to the other types of models tested in this thesis.
- Could not conclude that AI models would be recommended for misinformation detection as these models had an average classification accuracy of 64%.
- Also concluded that AI models provide faulty explainability for trustworthiness scores.
- Suggested that AI models should not be used by public users for misinformation detection but can be used for research purposes for further studies.

To synthesize, all 3 components of this thesis helped provide a better understanding of the strengths and limitations of each type of model used for misinformation detection.

7.3 Contribution to Society and Stakeholders

The significance of the findings of the thesis to society and the intended stakeholders provide the impacts described below.

Society:

- Provided a big data solution to help protect society against the harmful effects of misinformation through more accurate trustworthiness scoring of news articles.
- Proposed an data tool that can determine how trustworthy a given news article is, whether originated from mainstream, independent, or social media news outlets.
- Included explainability to build trust in the tool with the user.
- Warned that AI models should not be depended upon by public users for misinformation detection as responses can tend to be unreliable, which could impact people's perspectives on popular generative AI models.

Stakeholders:

- **Government agencies:** an added data tool to help invest in combating misinformation spread through harmful fake news.
- **Military agencies:** a data tool that can also combat misinformation that can impact business decisions, which in turn can disrupt crucial supply chain operations for the military.
- **Industry programmers:** a tool that can rapidly populate accurate trustworthiness scores for databases of news article texts with explainability metrics.
- **Public users:** an open-source tool that can be used to holistically determine if news found on a diversity of news outlets is trustworthy or not.

7.4 Future Work and Recommendations

Future work will likely require expanding the dimension vocabularies to include recent verified instances of misinformation including specified phrases, words, and punctuation. However, compiling these extensive vocabularies would require a lot of time and research in

order to not omit important indicative features. Yet, this improvement to the thesis work would help immensely with the quality of the finished product. Other work would also include training the base LLM on more news articles and potentially including non-English-language articles and dimension vocabularies. After finalizing the vocabularies, peer review is welcome to evaluate and certify the tool to be ready for open-source and government use. There can also be work done to adapt the tool to accommodate social media data; many features may have to be altered, but social media misinformation's impacts are just as profound as news misinformation's. Therefore, having a similar tool for social media would be just as practical for analysis and comprehensive trustworthiness predictions.

Regarding the limitation that association rule mining raised, future work can also include using association rule mining and the unsupervised learning methods used in this thesis to differentiate verified true and false statements. As mentioned at the end of Chapter 4, work has already been completed on this, however this work was not able to be included in the thesis, with the exception of some data visualizations that are shown in Chapter 10, because this work was done on a completely different dataset. Thereby it would digress from the conclusions drawn from comparing the LLMs, unsupervised learning methods, and AI models being performed on the news article dataset. This work would still help contribute to automated fact-checking research, which is equally instrumental in field research on misinformation.

As for recommendations for future research, there can be several encouragements given the promising results of this thesis. First, despite the insufficient results that the AI models provided from the evaluation and some previous studies from the literature review, since these models are fairly recent there still needs to be additional comprehensive studies on their ability to detect misinformation conducted, both in accuracy and in explainability. The prompts for the generative AI models can likewise be improved through prompt engineering to produce more trustworthiness scores and explainability for given article texts so as to not

lead to a loss of information in analyzing these models' efficacy in misinformation detection. While adjustments through adding clarifications to the prompts can produce more results, it should be noted that the AI models may still not be able to capture all context between the prompt and article text combined if there is too much complexity, leading to responses that may not be able to answer all aspects of the prompt. Therefore, keeping prompts relatively concise while adding enough specified context to make the prompts as effective as possible will be key to producing better results.

Next, since LLMs have been consistently accurate in determining trustworthiness scores for news articles, other supervised learning models should likewise be evaluated (i.e., simple machine learning models, neural networks) in their ability to detect misinformation. If these models also stand, they can likewise be recommended for viable misinformation detection research. This can also apply to other unsupervised learning models. Recommended work would also include re-evaluating the NLP dimension weights and vocabularies as well as conducting more studies on generative AI models' use for misinformation detection. In the future, misinformation detection research should incorporate communication dimensions to improve accuracy, as this would be an effective intersection between the fields of data science and communications to provide an interdisciplinary solution to the major societal problem of misinformation.

7.5 Final Thoughts

The main takeaway from this thesis is that given the vast amount of research done, and research gaps, on misinformation detection methods, it is important to resolve which methods, models, and/or tools are most effective at completing the crucial task of differentiating which is misinformation and which is not. The news is an important part of free speech and journalism in society, however misinformation is an abuse of free speech and can endanger reputations of citizens, organizations, and countries by spreading misleading information to

its readers. The ideal solution to the problem of misinformation is widespread media literacy and legislative regulations, however current solutions have required involvement in the data science field which has led to the creation of tools and models to combat misinformation. Target stakeholders likewise need to have awareness that such tools exist in order to be educated on which news is true or fake, with technological accountability to establish trust in these tools. Despite the harrowing task of winning the information war, data science has a major role to play in winning this war and to ensure the trustworthiness and effectiveness of tools past, present, and future.

Chapter 8

Acknowledgements

I would like to express my deepest gratitude to the following individuals and organizations who have played a significant role in the completion of this master's thesis.

First, I am very grateful to my thesis committee chair and advisor, Dr. David Ebert, for providing me with the utmost support throughout this research process. I would also like to extend my sincere appreciation to Dr. Gopi Danala and Dr. Wolfgang Jentner for providing crucial advising and feedback for this thesis.

Second, I would like to thank the members of my thesis committee. I greatly appreciate Dr. Dean Hougen and Dr. Naveen Kumar for their support and for helping shape the data-driven narrative and components of this research. As misinformation is not just a data science issue and is likewise closely related to the fields of communications and journalism, I am deeply grateful for also having Dr. Jeong-Nam Kim and Dr. Katerina Tsetsura from the Gaylord College of Journalism and Mass Communication on the committee. Their expertise and steadfast support have really helped in providing quality, transdisciplinary work for this thesis.

Finally and most importantly, I would like to thank the Data Institute for Societal Challenges (DISC) here at the University of Oklahoma. I am deeply grateful to DISC for helping financially support this research and for providing me with a graduate research assistantship that has become invaluably crucial to my research experience and development as a data

scientist. This assistantship has also relieved many financial burdens associated with pursuing a master's degree, and I am profoundly thankful for that as without this support, this thesis would not have been possible.

Chapter 9

References

[1] Adel, H., Dahou, A., Mabrouk, A., Abd Elaziz, M., Kayed, M., El-Henawy, I. M., ... & Amin Ali, A. (2022). Improving crisis events detection using distilbert with hunger games search algorithm. *Mathematics, 10*(3), 447.

[2] Akhtar, P., Ghouri, A. M., Khan, H. U. R., Amin ul Haq, M., Awan, U., Zahoor, N., ... & Ashraf, A. (2023). Detecting fake news and disinformation using artificial intelligence and machine learning to avoid supply chain disruptions. *Annals of Operations Research, 327*(2), 633-657.

[3] Amer, E., Kwak, K. S., & El-Sappagh, S. (2022). Context-based fake news detection model relying on deep learning models. *Electronics, 11*(8), 1255.

[4] Azzimonti, M., & Fernandes, M. (2023). Social media networks, fake news, and polarization. *European journal of political economy, 76*, 102256.

[5] Bhat, M. M. (2022). *Study of Effectiveness of Stylometry in Misinformation Detection* (Doctoral dissertation, The Ohio State University).

[6] Calvillo, D. P., Garcia, R. J., Bertrand, K., & Mayers, T. A. (2021). Personality factors and self-reported political news consumption predict susceptibility to political fake news. *Personality and individual differences, 174*, 110666.

[7] Caramancion, K. M. (2023). News Verifiers Showdown: A Comparative Performance Evaluation of ChatGPT 3.5, ChatGPT 4.0, Bing AI, and Bard in News Fact-Checking. *arXiv preprint arXiv:2306.17176*.

[8] Caruana, R., & Niculescu-Mizil, A. (2006, June). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning* (pp. 161-168).

[9] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR), 41*(3), 1-58.

[10] Chiu, M. M., Morakhovski, A., Ebert, D., Reinert, A., & Snyder, L. S. (2023). Detecting COVID-19 fake news on Twitter: Followers, emotions, relationships, and uncertainty. *American Behavioral Scientist*, 00027642231174329.

[11] Chiu, M. M., & Oh, Y. W. (2021). How fake news differs from personal lies. *American behavioral scientist, 65*(2), 243-258.

[12] Chiu, M. M., Park, C. H., Lee, H., Oh, Y. W., & Kim, J. N. (2022). Election Fraud and Misinformation on Twitter: Author, Cluster, and Message Antecedents. *Media and Communication, 10*(2), 66-80.

[13] Cui, Z. (2022, January). COVID-19 Fake News and Misinformation Detection using Transformer Learning. In *2022 3rd International Conference on Education, Knowledge and Information Management (ICEKIM)* (pp. 965-968).

[14] Cunha, B., & Manikonda, L. (2022). Classification of Misinformation in New Articles using Natural Language Processing and a Recurrent Neural Network. *arXiv preprint arXiv:2210.13534*.

[15] Dame Adjin-Tettey, T. (2022). Combating fake news, disinformation, and misinformation: Experimental evidence for media literacy education. *Cogent arts & humanities,*

9(1), 2037229.

[16] De Magistris, G., Russo, S., Roma, P., Starczewski, J. T., & Napoli, C. (2022). An explainable fake news detector based on named entity recognition and stance classification applied to covid-19. *Information, 13*(3), 137.

[17] Deng, J., & Lin, Y. (2022). The benefits and challenges of ChatGPT: An overview. *Frontiers in Computing and Intelligent Systems, 2*(2), 81-83.

[18] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[19] Duan, Y., Edwards, J. S., & Dwivedi, Y. K. (2019). Artificial intelligence for decision making in the era of Big Data–evolution, challenges and research agenda. *International journal of information management, 48*, 63-71.

[20] Fernando, A., & Wijayasiriwardhane, T. K. (2020, September). Identifying religious extremism-based threats in SriLanka using bilingual social media intelligence. In *2020 International Research Conference on Smart Computing and Systems Engineering (SCSE)* (pp. 103-110). IEEE.

[21] Gradoń, K. T., Hołyst, J. A., Moy, W. R., Sienkiewicz, J., & Suchecki, K. (2021). Countering misinformation: A multidisciplinary approach. *Big Data & Society, 8*(1), 20539517211013848.

[22] Grignolio, A., Morelli, M., & Tamietto, M. (2022). Why is fake news so fascinating to the brain?. *European Journal of Neuroscience, 56*(11), 5967-5971.

[23] Guess, A., Nyhan, B., & Reifler, J. (2018). Selective exposure to misinformation: Evidence from the consumption of fake news during the 2016 US presidential campaign.

[24] Guimarães, N., Figueira, Á., & Torgo, L. (2021). Can fake news detection models maintain the performance through time? A longitudinal evaluation of twitter publications. *Mathematics, 9*(22), 2988.

[25] Gunel, B., Du, J., Conneau, A., & Stoyanov, V. (2020). Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

[26] Hameleers, M., & Brosius, A. (2022). You are wrong because I am right! The perceived causes and ideological biases of misinformation beliefs. *International Journal of Public Opinion Research, 34*(1), edab028.

[27] Haupt, M. R., Li, J., & Mackey, T. K. (2021). Identifying and characterizing scientific authority-related misinformation discourse about hydroxychloroquine on twitter using unsupervised machine learning. *Big Data & Society, 8*(1), 20539517211013843.

[28] Hoes, E., Altay, S., & Bermeo, J. (2023). Using ChatGPT to Fight Misinformation: ChatGPT Nails 72% of 12,000 Verified Claims.

[29] Islam, M. R., Liu, S., Wang, X., & Xu, G. (2020). Deep learning for misinformation detection on online social networks: a survey and new perspectives. *Social Network Analysis and Mining, 10*, 1-20.

[30] Jamalzadeh, S., Barker, K., González, A. D., & Radhakrishnan, S. (2022). Protecting infrastructure performance from disinformation attacks. *Scientific Reports, 12*(1), 12707.

[31] Jang, S. M., & Kim, J. K. (2018). Third person effects of fake news: Fake news regulation and media literacy interventions. *Computers in human behavior, 80*, 295-302.

[32] Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2019). Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey.

Multimedia Tools and Applications, 78, 15169-15211.

[33] Jwa, H., Oh, D., Park, K., Kang, J. M., & Lim, H. (2019). exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences, 9*(19), 4062.

[34] Kertysova, K. (2018). Artificial intelligence and disinformation: How AI changes the way disinformation is produced, disseminated, and can be countered. *Security and Human Rights, 29*(1-4), 55-81.

[35] Khanum, M., Mahboob, T., Imtiaz, W., Ghafoor, H. A., & Sehar, R. (2015). A survey on unsupervised machine learning algorithms for automation, classification and maintenance. *International Journal of Computer Applications, 119*(13).

[36] Khanzode, K. C. A., & Sarode, R. D. (2020). Advantages and disadvantages of artificial intelligence and machine learning: A literature review. *International Journal of Library & Information Science (IJLIS), 9*(1), 3.

[37] Kochhar, S. K., & Kaur, R. (2023). Breaking the Taboos: Deploying Knowledge Differentiation to study COVID-19 ramifications on Women's Menstrual Health. *International Journal of Computing and Digital Systems, 13*(1), 1-1.

[38] Kong, S. H., Tan, L. M., Gan, K. H., & Samsudin, N. H. (2020, April). Fake news detection using deep learning. In *2020 IEEE 10th symposium on computer applications & industrial electronics (ISCAIE)* (pp. 102-107). IEEE.

[39] Kumbhare, T. A., & Chobe, S. V. (2014). An overview of association rule mining algorithms. *International Journal of Computer Science and Information Technologies, 5*(1), 927-930.

[40] Kwek, A., Peh, L., Tan, J., & Lee, J. X. (2023). Distractions, analytical thinking and falling for fake news: A survey of psychological factors. *Humanities and Social*

Sciences Communications, 10(1), 1-12.

[41] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In *2008 eighth ieee international conference on data mining* (pp. 413-422). IEEE.

[42] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

[43] Martel, C., Pennycook, G., & Rand, D. G. (2020). Reliance on emotion promotes belief in fake news. *Cognitive research: principles and implications, 5*, 1-20.

[44] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics, 5*, 115-133.

[45] Meyer, J. G., Urbanowicz, R. J., Martin, P. C., O'Connor, K., Li, R., Peng, P. C., ... & Moore, J. H. (2023). ChatGPT and large language models in academia: opportunities and challenges. *BioData Mining, 16*(1), 20.

[46] Mhatre, S., & Masurkar, A. (2021, June). A hybrid method for fake news detection using cosine similarity scores. In *2021 International Conference on Communication information and Computing Technology (ICCICT)* (pp. 1-6). IEEE.

[47] Min, E., Rong, Y., Bian, Y., Xu, T., Zhao, P., Huang, J., & Ananiadou, S. (2022). Divide-and-Conquer: Post-User Interaction Network for Fake News Detection on Social Media. In *Proceedings of the ACM Web Conference 2022* (pp. 1148-1158).

[48] Ng, L. H., & Taeihagh, A. (2021). How does fake news spread? Understanding pathways of disinformation spread through APIs. *Policy & Internet, 13*(4), 560-585.

[49] Oh, Y. W., & Park, C. H. (2021). Machine cleaning of online opinion spam: Developing a machine-learning algorithm for detecting deceptive comments. *American behavioral scientist, 65*(2), 389-403.

[50] Oliva, C., Palacio-Marín, I., Lago-Fernández, L. F., & Arroyo, D. (2022, August). Rumor and clickbait detection by combining information divergence measures and deep learning techniques. In *Proceedings of the 17th International Conference on Availability, Reliability and Security* (pp. 1-6).

[51] Paredes, C. M. G., Machuca, C., & Claudio, Y. M. S. (2023). ChatGPT API: Brief overview and integration in Software Development. *International Journal of Engineering Insights, 1*(1), 25-29.

[52] Przybyła, P., & Soto, A. J. (2021). When classification accuracy is not enough: Explaining news credibility assessment. *Information Processing & Management, 58*(5), 102653.

[53] Rahutomo, F., Kitasuka, T., & Aritsugi, M. (2012, October). Semantic cosine similarity. In *The 7th international student conference on advanced science and technology ICAST* (Vol. 4, No. 1, p. 1).

[54] Rai, N., Kumar, D., Kaushik, N., Raj, C., & Ali, A. (2022). Fake News Classification using transformer based enhanced LSTM and BERT. *International Journal of Cognitive Computing in Engineering, 3*, 98-105.

[55] Ramadasa, I., Liyanage, L., Asanka, D., & Dilanka, T. (2022). Analysis of the effectiveness of using google translations api for nlp of sinhalese.

[56] Ray, P. P. (2023). ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems.*

[57] Reis, J. C., Correia, A., Murai, F., Veloso, A., & Benevenuto, F. (2019). Supervised learning for fake news detection. *IEEE Intelligent Systems, 34*(2), 76-81.

[58] Riego, N. C. R., & Villarba, D. B. (2023). Utilization of Multinomial Naive Bayes Algorithm and Term Frequency Inverse Document Frequency (TF-IDF Vectorizer) in Checking the Credibility of News Tweet in the Philippines. *arXiv preprint arXiv:2306.00018*.

[59] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

[60] Serrano, J. C. M., Papakyriakopoulos, O., & Hegelich, S. (2020, July). NLP-based feature extraction for the detection of COVID-19 misinformation videos on YouTube. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*.

[61] Seth, N. (2021, August 26). *Part 2: Topic modeling and Latent Dirichlet allocation (LDA) using Gensim and Sklearn*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/part-2-topic-modeling-and-latent-dirichlet-allocation-lda-using-gensim-and-sklearn/

[62] Shahi, G. K., & Nandini, D. (2020). FakeCovid–A multilingual cross-domain fact check news dataset for COVID-19. *arXiv preprint arXiv:2006.11343*.

[63] Sharma, D. K., Garg, S., & Shrivastava, P. (2021, February). Evaluation of tools and extension for fake news detection. In *2021 International Conference on Innovative Practices in Technology and Management (ICIPTM)* (pp. 227-232).

[64] Shinde, A. (2022, May). Unsupervised Detection of Misinformation in Financial Statements. In *The International FLAIRS Conference Proceedings* (Vol. 35).

[65] Shu, K., Mahudeswaran, D., & Liu, H. (2019). FakeNewsTracker: a tool for fake news collection, detection, and visualization. *Computational and Mathematical Organization Theory, 25*, 60-71.

[66] Su, Q., Wan, M., Liu, X., & Huang, C. R. (2020). Motivations, methods and metrics of misinformation detection: an NLP perspective. *Natural Language Processing Research, 1*(1-2), 1-13.

[67] Tamkin, A., Brundage, M., Clark, J., & Ganguli, D. (2021). Understanding the capabilities, limitations, and societal impact of large language models. *arXiv preprint arXiv:2102.02503*.

[68] Thorne, J., Vlachos, A., Christodoulopoulos, C., & Mittal, A. (2018). FEVER: a large-scale dataset for fact extraction and VERification. *arXiv preprint arXiv:1803.05355*.

[69] Titiliuc, C., Ruseti, S., & Dascalu, M. (2020, September). What's Been Happening in the Romanian News Landscape? A Detailed Analysis Grounded in Natural Language Processing Techniques. In *2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)* (pp. 195-201). IEEE.

[70] Tucker, Joshua A., et al. "Social media, political polarization, and political disinformation: A review of the scientific literature." *Political polarization, and political disinformation: a review of the scientific literature (March 19, 2018)* (2018).

[71] Turner, J., Kantardzic, M., Vickers-Smith, R., & Brown, A. G. (2023). Detecting Tweets Containing Cannabidiol-Related COVID-19 Misinformation Using Transformer Language Models and Warning Letters From Food and Drug Administration: Content Analysis and Identification. *JMIR infodemiology, 3*(1), e38390.

[72] van de Meerakker, K. O. (2022). The foreign language effect on the credibility of fake news messages among Dutch news readers and the influence of emotional loading.

[73] Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research, 9*(11).

[74] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems, 30*.

[75] Verma, P. K., Agrawal, P., Amorim, I., & Prodan, R. (2021). WELFake: word embedding over linguistic features for fake news detection. *IEEE Transactions on Computational Social Systems, 8*(4), 881-893.

[76] Wadud, M. A. H., Mridha, M. F., & Rahman, M. M. (2022). Word embedding methods for word representation in deep learning for natural language processing. *Iraqi Journal of Science*, 1349-1361.

[77] Waisberg, E., Ong, J., Masalkhi, M., Zaman, N., Sarker, P., Lee, A. G., & Tavakkoli, A. (2023). Google's AI chatbot "Bard": a side-by-side comparison with ChatGPT and its utilization in ophthalmology. *Eye*, 1-4.

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

[79] Watts, D. J., Rothschild, D. M., & Mobius, M. (2021). Measuring the news and its impact on democracy. *Proceedings of the National Academy of Sciences, 118*(15), e1912443118.

[80] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

[81] Zhang, C., Gupta, A., Kauten, C., Deokar, A. V., & Qin, X. (2019). Detecting fake news for reducing misinformation risks using analytics approaches. *European Journal of Operational Research, 279*(3), 1036-1052.

[82] Zhang, X. Examining COVID-19 Vaccination Misinformation and Clarification by the Public Sector in Hong Kong.

[83] Zhong, H., & Mei, H. (2017). An empirical study on API usages. IEEE Transactions on Software Engineering, 45(4), 319-334.

Chapter 10

Appendices

10.1 Appendix A: Supplementary Code Snippets

10.1.1 NLP Dimension Vocabularies

```
# 1. Sentiment
sentiment_list = ['amazing', 'fantastic', 'excellent', 'good', 'wonderful', 'great',
                  'outstanding', 'terrific', 'phenomenal', 'remarkable',
                  'glorious', 'delightful', 'superb', 'brilliant', 'spectacular',
                  'extraordinary', 'marvelous', 'awesome', 'heartwarming', 'inspiring',
                  'inspired', 'terrible', 'awful', 'horrible', 'disastrous', 'tragic',
                  'catastrophic', 'devastating', 'abysmal', 'atrocious', 'disgusting',
                  'disturbing', 'appalling', 'dreadful', 'repulsive', 'wretched',
                  'repugnant', 'revolting', 'disgraceful', 'shameful', 'hateful',
                  'love', 'hate', 'dangerous', 'happy', 'sad', 'bad', 'positive',
                 'negative', 'worst', 'badly', 'joyful', 'joyous', 'thrilled',
                 'miserable', 'misery', 'depressing', 'worried', 'angry', 'ominous',
                 'chilling', 'evil', 'positively', 'negatively']
```

Figure 10.1: Sentiment Dimension Vocabulary

```
# 2. Persuasion/Bias
persuasion_list = ['clearly', 'obviously', 'certainly', 'undoubtedly', 'absolutely',
                   'must', 'should', 'ought', 'need', 'will', 'always', 'never',
                   'only', 'everyone', 'nobody', 'you', 'we', 'our', 'your',
                   'cannot', 'no', 'demand', 'surely', 'my', 'us', 'insist',
                  'ignore', 'do', 'neglect', 'sure']
```

Figure 10.2: Persuasion Dimension Vocabulary

```
# 3. Exaggeration
exaggeration_list = ['unbelievable', 'astonishing', 'mind-blowing', 'epic',
                     'unprecedented', 'outrageous', 'outrage', 'tremendous',
                     'enormous', 'massive', 'colossal', 'gigantic', 'huge',
                     'immense', 'unimaginable', 'spectacular', '!', 'shocking',
                     'scandal', 'scandalous', 'repulsive', 'unacceptable',
                     'horrific', 'offensive', 'outraged', 'furious', 'enraged',
                     'enrages', 'angers', 'angered', 'incensed', 'doom', 'end',
                     'downfall', 'corrupt', 'banana', 'circus', 'crazy', 'stupid',
                     'lot', 'load', 'slams', 'slammed', 'trolling', 'amateur',
                    'amateurs', 'fake', 'blasts', 'blasted', 'cover-up',
                    'outlandish', 'sensational', 'insane', 'bizarre', 'incredible',
                    'wild', 'deranged', 'deluded', 'delusion', 'delusions', 'sick',
                    'garbage', 'vicious', 'viciously', 'attack', 'attacked',
                    'mean', 'hell', 'demented', 'wildly', 'conspiring']
```

Figure 10.3: Exaggeration Dimension Vocabulary

```
# 4. Context
context_list = ['"', '‘', '“', '’', '”']
```

Figure 10.4: Context Dimension Vocabulary

```
# 5. Inclusion of Multiple Perspectives
multiple_perspectives_list = ['said', 'says', 'claims', 'claim', 'say', 'claimed', 'remark', 'remarked',
                              'state', 'states', 'stated', 'assert', 'asserts', 'asserted',
                              'declare', 'declares', 'declared', 'express', 'expressed',
                              'mention', 'mentioned', 'mentions', 'announce', 'announced',
                              'allege', 'alleged', 'insist', 'insisted', 'insists', 'propose',
                              'proposed', 'suggest', 'suggested', 'view', 'perspective', 'insight',
                              'advice', 'opinion', 'argument', 'clear', 'contend', 'contends', 'contended',
                             'debated', 'discussed', 'disputed', 'differ', 'disputes', 'disagreed',
                             'disagrees', 'disagree', 'agreed', 'agree', 'agrees', 'interpret',
                             'interprets', 'interpreted']
```

Figure 10.5: Multiple Perspectives Dimension Vocabulary

```
# 7. Use of Statistics
statistics_list = ['%', '$', 'poll', 'polls', 'percent', 'trend',
                   'studies', 'study', 'statistics', 'data', 'survey',
                   'surveys', 'decline', 'increase', 'record', 'margin',
                   'trends', 'trending', 'declined', 'percent', 'percentage',
                   'number', 'numeral', 'data', 'statistic', 'year']
```

Figure 10.6: Statistics Dimension Vocabulary

```
# 8. Referencing of Previous Articles
previous_articles_list = ['op-ed', 'report', 'reported', 'article', 'editorial',
                          'coverage', 'story', 'write-up', 'piece', 'segment',
                          'publication', 'news', 'reports', 'recap', 'opinion',
                          'column']
```

Figure 10.7: Previous Articles Dimension Vocabulary

```
# 10. Distraction
distraction_list = ['distraction', 'distract', 'diversion', 'divert', 'misdirection',
                    'herring', 'mirrors', 'sideshow', 'smokescreen', 'deflection', 'deflect',
                    'divisive', 'divide', 'tactic', 'misleading', 'mislead', 'tangential',
                    'false', 'misplaced', 'inconsistent', 'but', 'confuse',
                    'confusing', 'confused', 'misinformation', 'misinform', 'misinformed',
                    'deflected', 'deceiving', 'deception', 'deceitful', 'fabrication',
                   'fabricate', 'fabricated', 'distorted', 'manipulate', 'manipulative',
                   'manipulation', 'conceal', 'propaganda']
```

Figure 10.8: Distraction Dimension Vocabulary

```
# 11. Verification of Claims
verification_list = ['alleged', 'unverified', 'unsubstantiated', 'unconfirmed', 'confirm', 'speculative',
                     'speculation', 'rumored', 'rumors', 'rumor', 'false', 'true', 'truth', 'doubtful', 'unproven',
                     'questionable', 'verified', 'verify', 'confirmed', 'proven', 'authenticated',
                     'substantiated', 'validated', 'validate', 'validates', 'established', 'factual', 'facts', 'corroborated',
                     'corroborates', 'corroborate', 'reliable', 'sources', 'alleges', 'substantiates',
                     'verifies', 'establishes', 'establish', 'source', 'proves', 'disproves', 'disproven',
                     'comment', 'baseless']
```

Figure 10.9: Claim Verification Dimension Vocabulary

```
# 12. Logical Coherence
logical_coherence_list = ['therefore', 'because', 'thus', 'hence', 'consequently', 'moreover', 'furthermore',
                          'additionally', 'nevertheless', 'regardless', 'however', 'nonetheless', 'contrast',
                          'subsequently', 'consequence', 'accordingly', 'so', 'then', 'considering', 'resulting',
                          'due', 'also', 'likewise', 'similarly', 'summarize', 'summary', 'conclusion', 'result',
                          'conclude']
```

Figure 10.10: Logical Coherence Dimension Vocabulary

```
# 13. Clickbait Title
clickbait_list = ['unbelievable', 'astonishing', 'mind-blowing', 'unprecedented', 'outrageous',
                  'unimaginable', 'spectacular', 'shocking', 'scandal', 'scandalous', 'fake',
                  'sensational', 'jaw-dropping', 'insane', 'revealed', 'must-see', 'exclusive',
                  'bizarre', 'secret', 'unveiled', 'controversial', 'controversy', 'exposed',
                  'incredible', 'mind-bending', 'amazing', 'fantastic', 'excellent',
                  'outstanding', 'terrific', 'phenomenal', 'glorious', 'brilliant', 'spectacular',
                  'extraordinary', 'marvelous', 'awesome', 'heartwarming', 'inspiring',
                  'terrible', 'awful', 'horrible', 'disastrous',
                  'catastrophic', 'devastating', 'abysmal', 'atrocious', 'disgusting',
                  'disturbing', 'appalling', 'dreadful', 'repulsive',
                  'repugnant', 'revolting', 'disgraceful', 'shameful', 'hateful',
                  'dangerous', 'worst', 'thrilling', 'miserable', 'misery', 'depressing',
                  'worried', 'angry', 'ominous', 'chilling', 'evil']
```

Figure 10.11: Clickbait Title Dimension Vocabulary

10.1.2 Data Tool Functions

```
# Function to count and normalize count of times dimension vocabulary appears in tokenized text
# This can be used for most dimensions
def count_normalize_vocab(article_text, vocab_list):
    # Tokenize the text using NLTK's word_tokenize function
    tokenized_article = word_tokenize(article_text)

    # Convert all tokens to lowercase for case-insensitive comparison
    tokenized_article_lower = [token.lower() for token in tokenized_article]

    # Initialize the counter variable
    total_count = 0

    # Loop through each token in the tokenized article
    for token in tokenized_article_lower:
        # Check if the lowercase version of the token is in the vocab list
        if token in vocab_list:
            # Increment the counter variable
            total_count += 1

    # Normalize count based on the length of the tokenize text
    normalized_count = total_count / len(tokenized_article)

    # Return the normalized count
    return normalized_count
```

Figure 10.12: Function to Count and Normalize Vocabulary

```python
# Function to extract named entities from article text
# This will create the vocabulary list for the named entities dimension
nlp = spacy.load("en_core_web_sm", disable=["tagger", "parser"])
named_entities = []
specific_labels = ['PERSON', 'NORP', 'FAC', 'ORG', 'GPE', 'EVENT', 'LAW', 'DATE']
def extract_named_entities(article_text, ner_model, allowed_labels=None):
    # Process the article text with spaCy NER model
    doc = ner_model(article_text)

    # Get the named entities from the processed document
    if allowed_labels is None:
        # If no allowed_labels provided, extract all named entities
        named_entities = [ent.text for ent in doc.ents]
    else:
        # Extract named entities with allowed_labels only
        named_entities = [ent.text for ent in doc.ents if ent.label_ in allowed_labels]

    # Return list of extracted named entities
    return named_entities
```

Figure 10.13: Function to Extract Named Entities

```python
# Normalize named entities
def count_normalize_named_entities(article_text, named_entities_list):
    # Tokenize the text using NLTK's word_tokenize function
    tokenized_article = word_tokenize(article_text)

    return len(named_entities_list) / len(tokenized_article)
```

Figure 10.14: Function to Count and Normalize Named Entities

```python
# Function to count and normalize count of numbers as well as statistics tokens
# Used for statistics dimension only
def count_normalize_statistics(article_text, stats_list):
    # Tokenize the text using NLTK's word_tokenize function
    tokenized_article = word_tokenize(article_text)

    # Convert all tokens to lowercase for case-insensitive comparison
    tokenized_article_lower = [token.lower() for token in tokenized_article]

    # Initialize the counter variable
    total_count = 0

    # Loop through each token in the tokenized article
    for token in tokenized_article_lower:
        # Check if the lowercase version of the token is in the statistics vocab list or is a number
        if token in stats_list or any(char.isdigit() for char in token):
            # Increment the counter variable
            total_count += 1

    # Normalize count based on the length of the tokenize text
    normalized_count = total_count / len(tokenized_article)

    # Return the normalized count
    return normalized_count
```

Figure 10.15: Function to Count and Normalize Statistics Count

```python
# Function to tf-idf vectorize and average the tf-idf of the article text
# Used for term frequency dimension only
def vectorize__normalize_tfidf(article_text):
    # Tokenize the article text, including punctuation
    tokens_with_punctuation = word_tokenize(article_text.lower())

    # Join the tokens back into a sentence
    tokenized_text = " ".join(tokens_with_punctuation)

    # Create a TfidfVectorizer object
    vectorizer = TfidfVectorizer()

    # Fit and transform the cleaned text to get the TF-IDF matrix
    tfidf_vector = vectorizer.fit_transform([tokenized_text])

    # Sum up the values in the TF-IDF vector
    tfidf_sum = np.sum(tfidf_vector.toarray()[0])

    # Normalize term frequency dimension by taking average
    normalized_tfidf = tfidf_sum / len(tfidf_vector.toarray()[0])

    return normalized_tfidf
```

Figure 10.16: Function to Normalize TF-IDF Vectors

```
# Function to calculate individual scores for the NLP dimensions
def individual_scores(sentiment, persuasion, exaggeration, context, multiple_perspectives, named_entities,
                      statistics, previous_articles, term_frequency, distraction, verification, logical_coherence,
                      clickbait_title):
    # Note that these thresholds were determined using the percentiles of the dimension distributions
    # First generate sentiment score (less sentiment = higher trust)
    if sentiment <= 0:
        sentiment_score = 10
    elif sentiment > 0 and sentiment <= 0.001:
        sentiment_score = 9
    elif sentiment > 0.001 and sentiment <= 0.0015:
        sentiment_score = 8
    elif sentiment > 0.0015 and sentiment <= 0.002:
        sentiment_score = 7
    elif sentiment > 0.002 and sentiment <= 0.0025:
        sentiment_score = 6
    elif sentiment > 0.0025 and sentiment <= 0.003:
        sentiment_score = 5
    elif sentiment > 0.003 and sentiment <= 0.0035:
        sentiment_score = 4
    elif sentiment > 0.0035 and sentiment <= 0.004:
        sentiment_score = 3
    elif sentiment > 0.004 and sentiment <= 0.0045:
        sentiment_score = 2
    elif sentiment > 0.0045 and sentiment <= 0.005:
        sentiment_score = 1
    elif sentiment > 0.005:
        sentiment_score = 0
```

Figure 10.17: Function to Set NLP Dimension Scores (Truncated)

```
# Generate verification of claims score (more verification = higher trust)
if verification > 0.003:
    verification_score = 4
elif verification > 0.002 and verification <= 0.003:
    verification_score = 3
elif verification > 0.001 and verification <= 0.002:
    verification_score = 2
elif verification > 0 and verification <= 0.001:
    verification_score = 1
else:
    verification_score = 0

# Generate previous articles score (more previous articles = higher trust)
if previous_articles > 0.001:
    previous_articles_score = 2
elif previous_articles > 0 and previous_articles <= 0.001:
    previous_articles_score = 1
else:
    previous_articles_score = 0

# Generate distraction score (less distraction = higher trust)
if distraction <= 0:
    distraction_score = 2
elif distraction > 0 and distraction <= 0.0025:
    distraction_score = 1
else:
    distraction_score = 0

# Finally generate clickbait title score (less clickbait = higher trust)
if clickbait_title <= 0:
    clickbait_title_score = 2
elif clickbait_title > 0 and clickbait_title <= 0.05:
    clickbait_title_score = 1
else:
    clickbait_title_score = 0

# Return all dimension scores
return sentiment_score, persuasion_score, exaggeration_score, context_score, multiple_perspectives_score, named_entities_score, statistics_score,
```

Figure 10.18: Function to Set NLP Dimension Scores (Continued, Truncated)

```
# Function to concatenate dimensions together to prepare as text input
def concat_dimensions(article_text, sentiment, persuasion, exaggeration, context, multiple_perspectives, named_entities,
                      statistics, previous_articles, term_frequency, distraction, verification,
                      logical_coherence, clickbait):
    concatenated_dimensions = (str(sentiment) + " " + str(persuasion) + " " + str(exaggeration) + " " + str(context) + " " + str(multiple_perspectives) +
                        " " + str(named_entities) + " " + str(statistics) + " " + str(previous_articles) + " " + str(term_frequency) + " " + str(distraction) +
                        " " + str(verification) + " " + str(logical_coherence) + " " + str(clickbait) + " " + article_text)
    return concatenated_dimensions
```

Figure 10.19: Function to Concatenate Dimensions

```
# Define maximum sequence length
max_seq_len = 160

# Load pre-trained tokenizer and RoBERTa base model
tokenizer = RobertaTokenizer.from_pretrained('roberta_model')
base_roberta_model = TFRobertaModel.from_pretrained('roberta-base')

# Initialize the prediction model with the same architecture as your fine-tuned model
inputs = tf.keras.layers.Input(shape=(max_seq_len,), dtype='int32', name='input_ids')
outputs = base_roberta_model(inputs)[0][:, 0, :]  # Select the [CLS] token representation
regression_head = tf.keras.layers.Dense(1, activation='linear')(outputs)
prediction_model = tf.keras.Model(inputs, regression_head)

# Load the fine-tuned weights
prediction_model.load_weights("roberta_model_weights.h5")

# Compile the model for prediction (optional if it was compiled during training)
prediction_model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=1e-5), loss='mean_squared_error', metrics=['mae'])
```

Figure 10.20: Importing the RoBERTa Weights and Compiling the Model

```
# Tokenize and pad the input using the fine-tuned tokenizer
input_ids = tokenizer([concatenated_text], padding='max_length', truncation=True, max_length=max_seq_len, return_tensors="tf")["input_ids"]

# Make prediction
prediction = prediction_model.predict(input_ids)
```

Figure 10.21: Making Predictions Using the RoBERTa Model

10.2 Appendix B: Additional Visualizations and Images

10.2.1 LLM Training Versus Validation

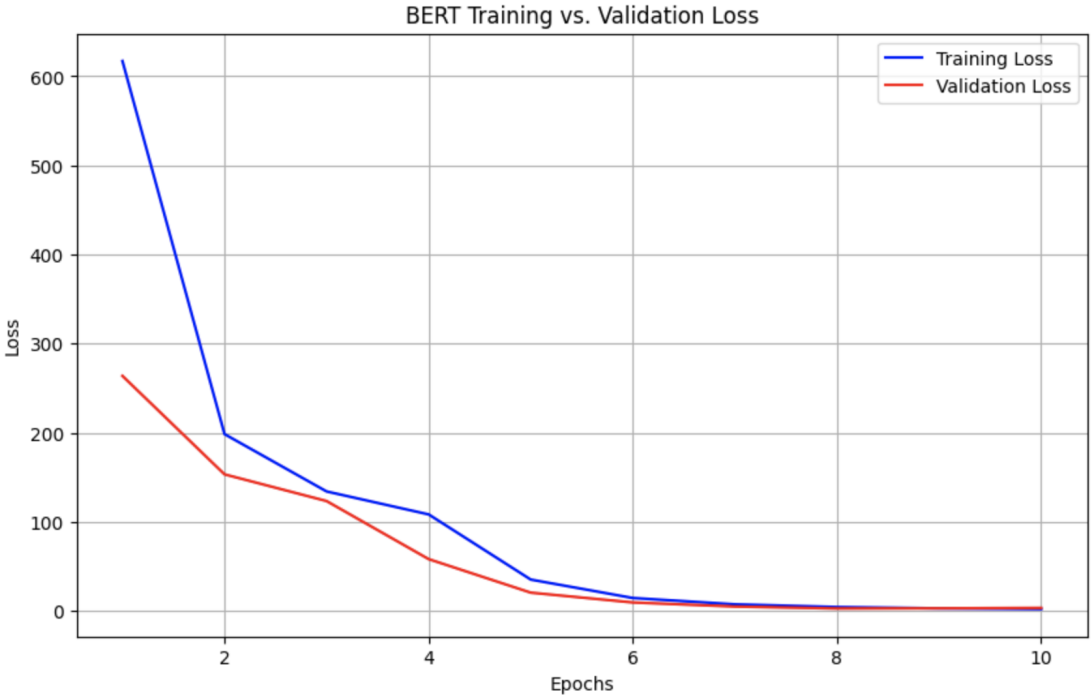


Figure 10.22: BERT Training vs. Validation Loss

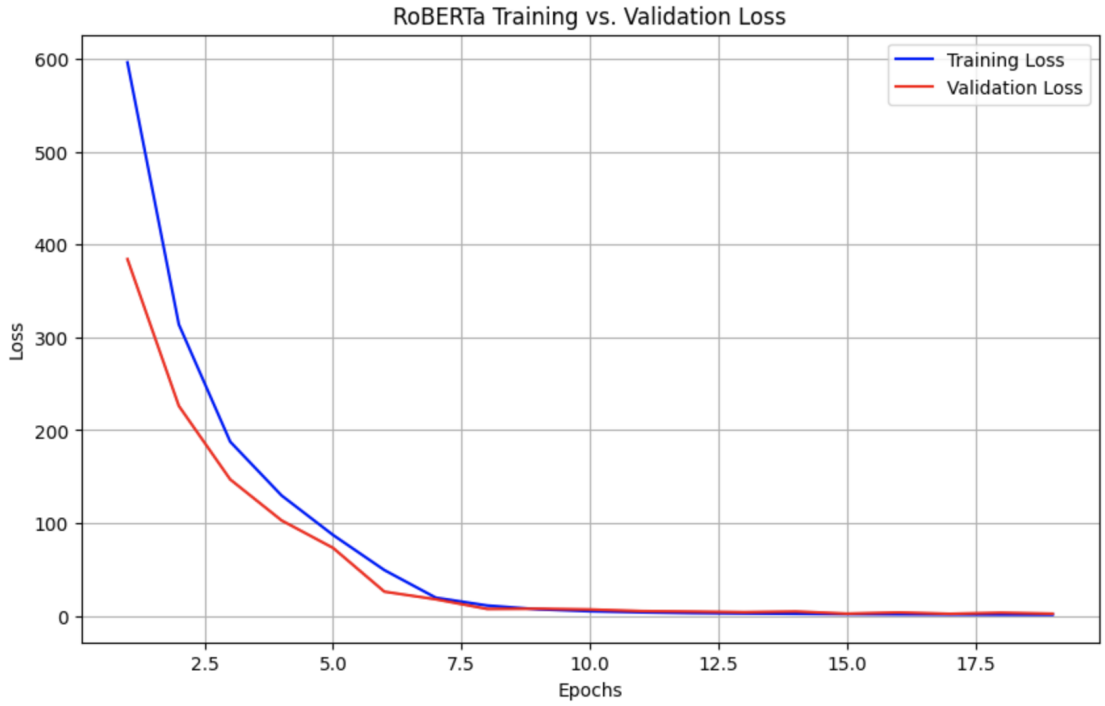


Figure 10.23: RoBERTa Training vs. Validation Loss

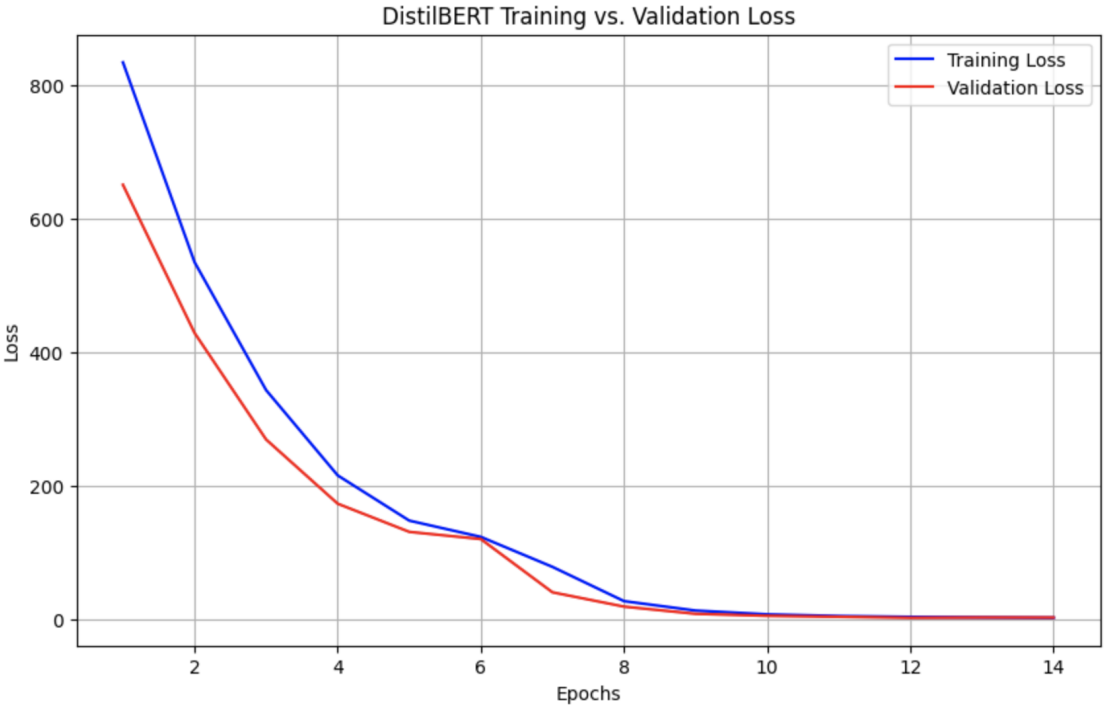


Figure 10.24: DistilBERT Training vs. Validation Loss

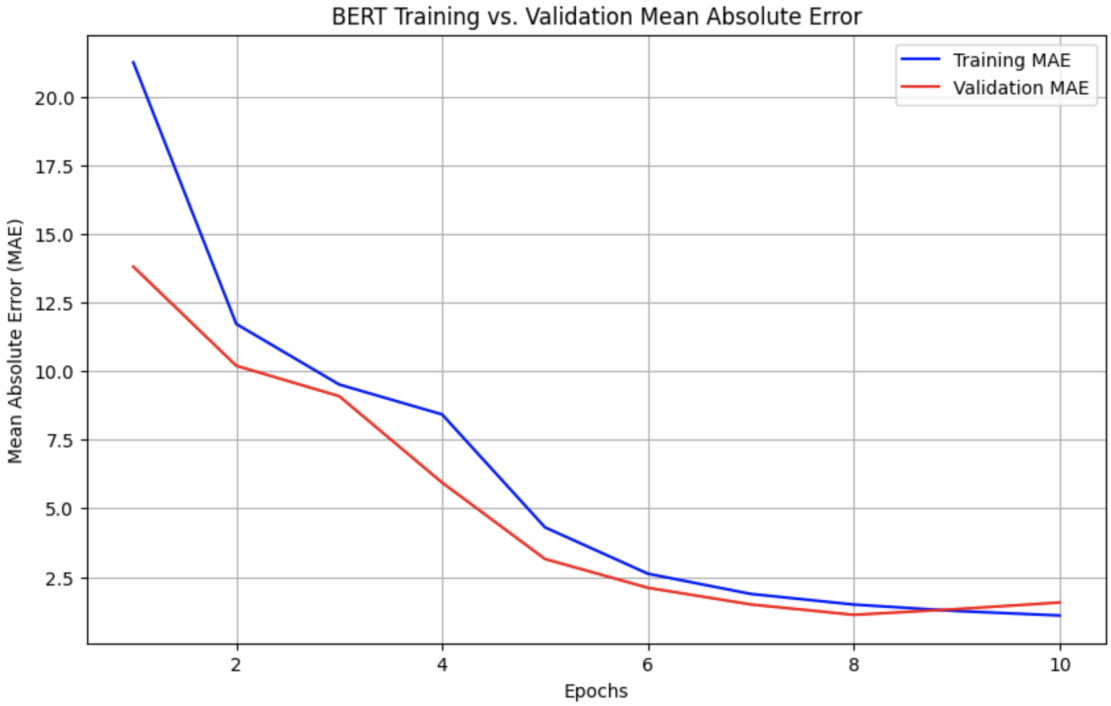


Figure 10.25: BERT Training vs. Validation Mean Absolute Error

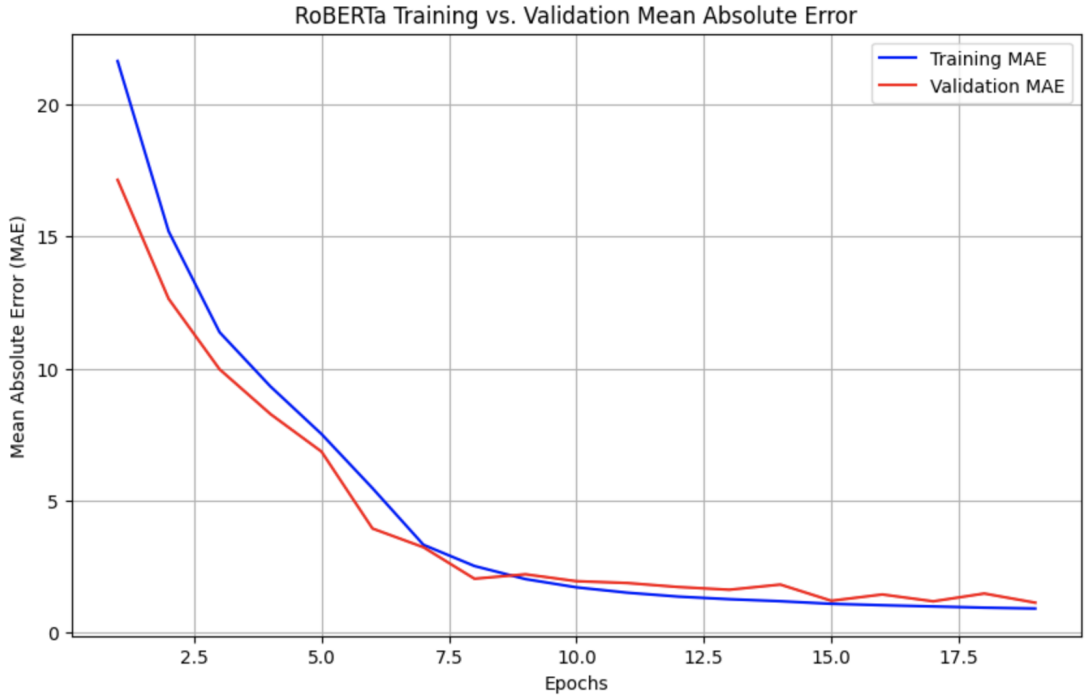


Figure 10.26: RoBERTa Training vs. Validation Mean Absolute Error

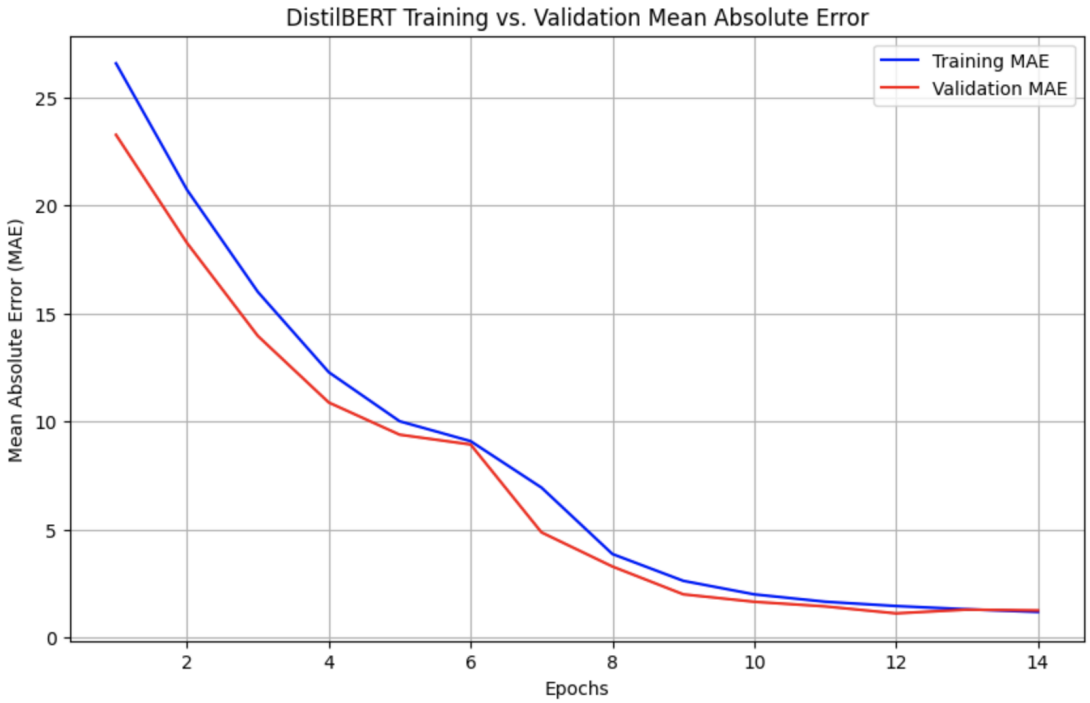


Figure 10.27: DistilBERT Training vs. Validation Mean Absolute Error

10.2.2 Association Rule Mining on Politifact Kaggle Dataset

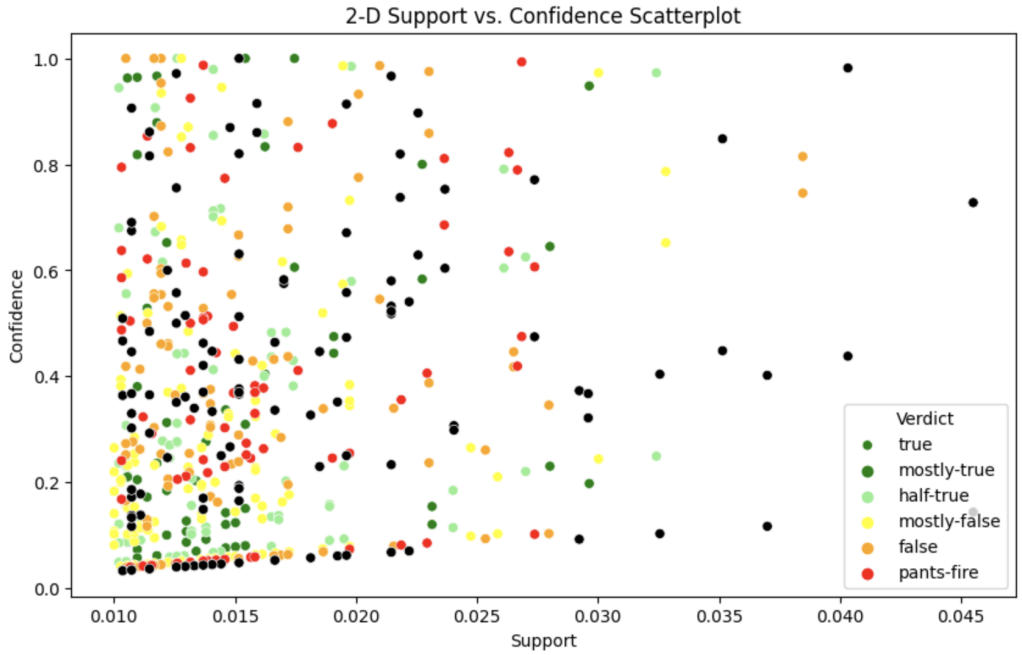


Figure 10.28: 2-D Scatter Plot of Politifact Kaggle Dataset Association Rules

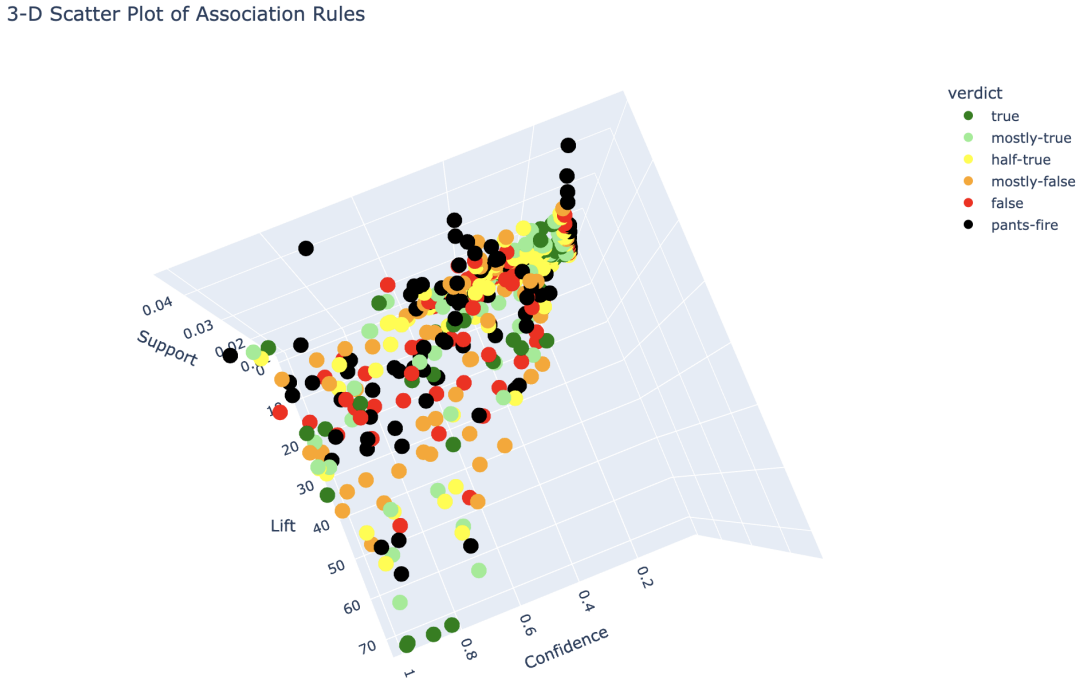


Figure 10.29: 3-D Scatter Plot of Politifact Kaggle Dataset Association Rules

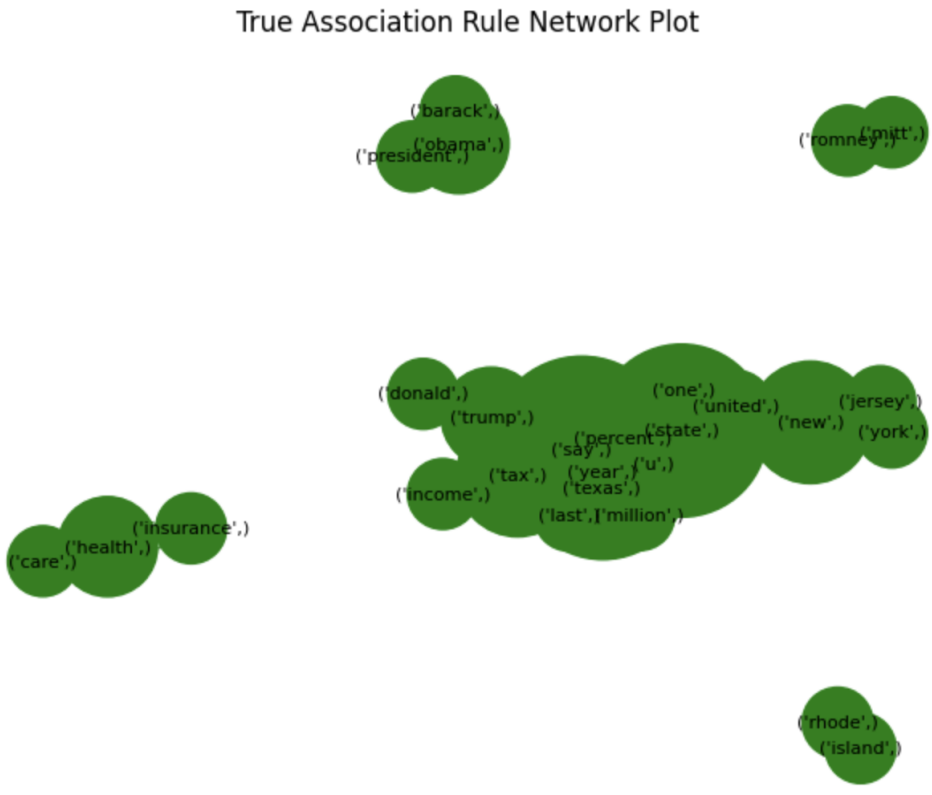


Figure 10.30: Network Plot of True Statement Association Rules

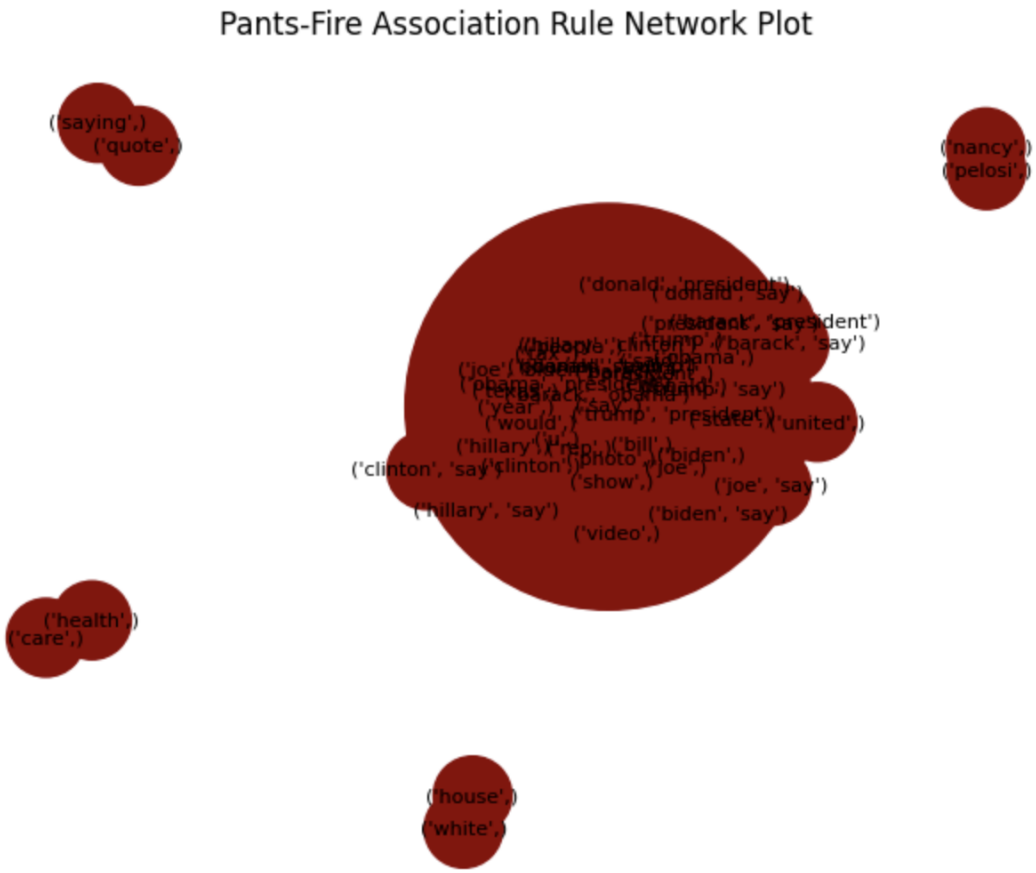


Figure 10.31: Network Plot of Pants-Fire Statement Association Rules

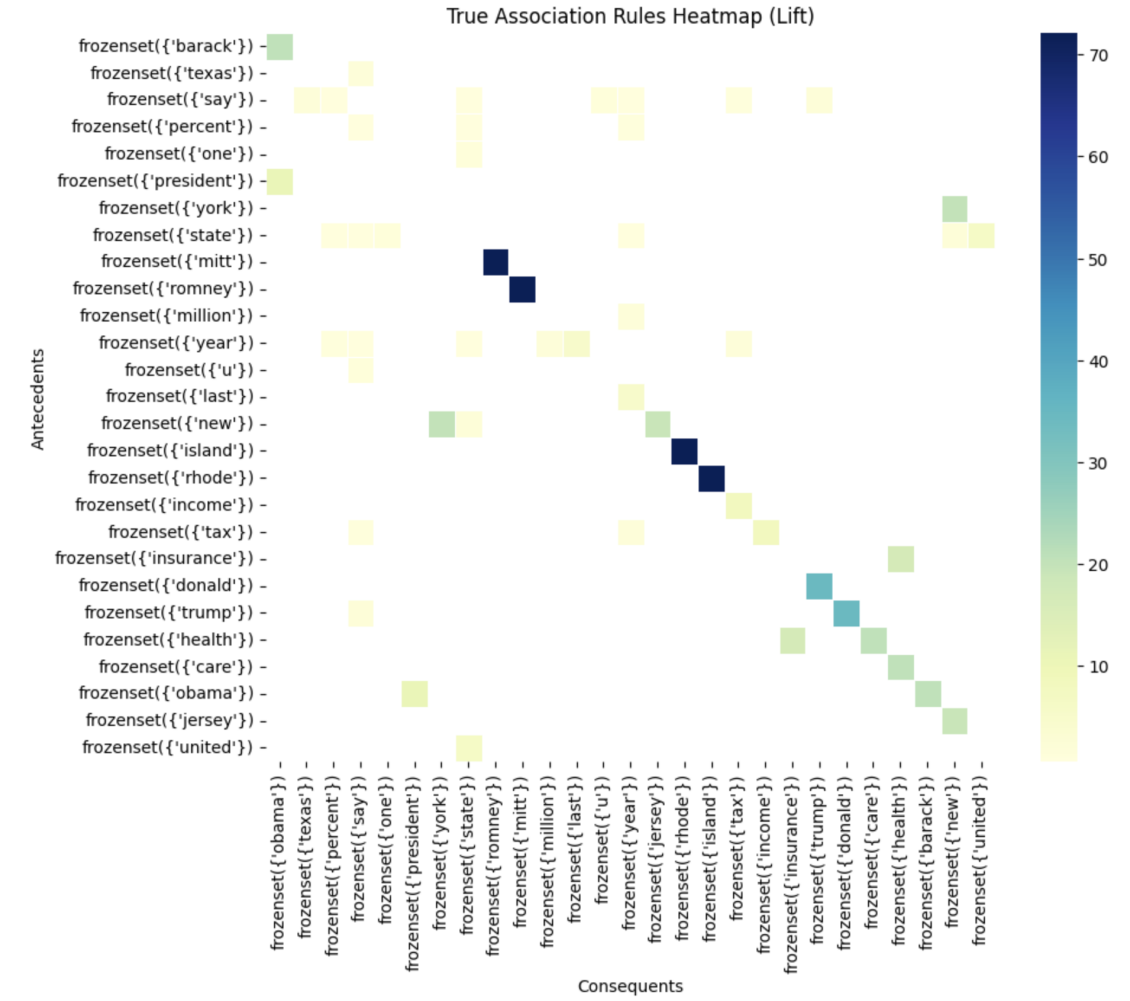


Figure 10.32: Heatmap of True Statement Association Rules

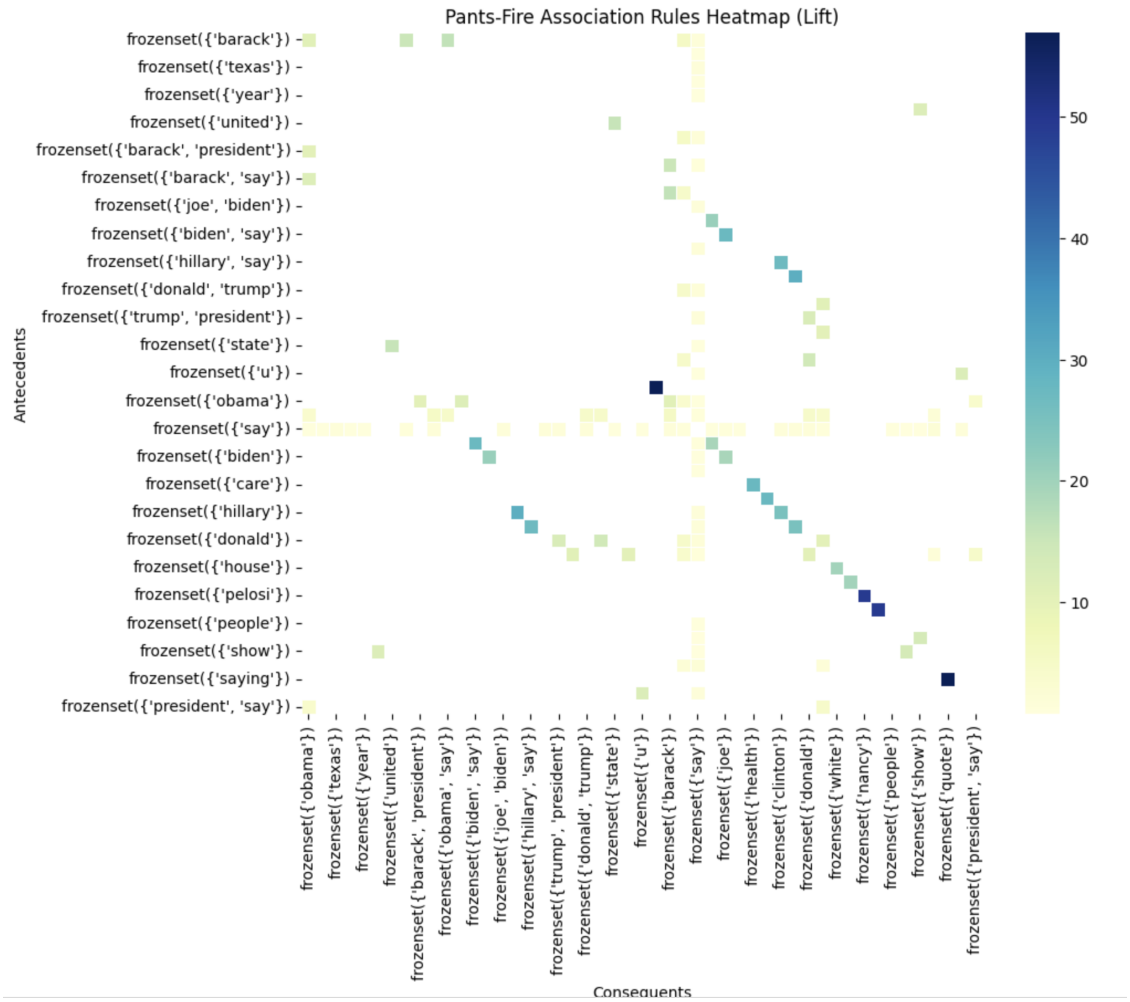


Figure 10.33: Heatmap of Pants-Fire Association Rules

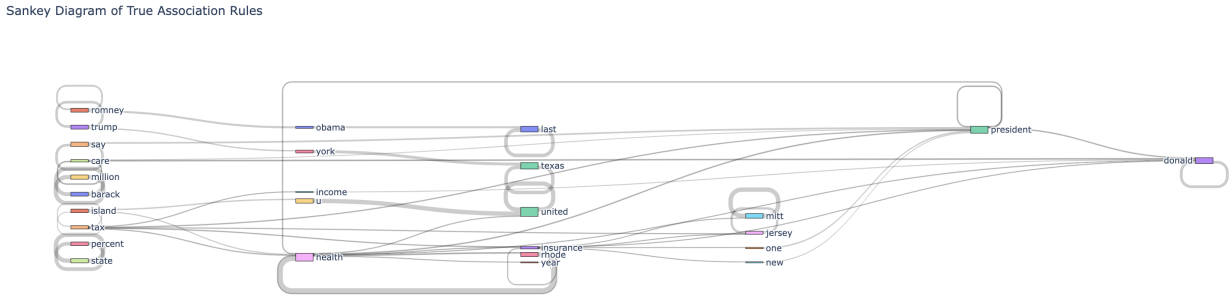


Figure 10.34: Sankey Diagram of True Statement Association Rules

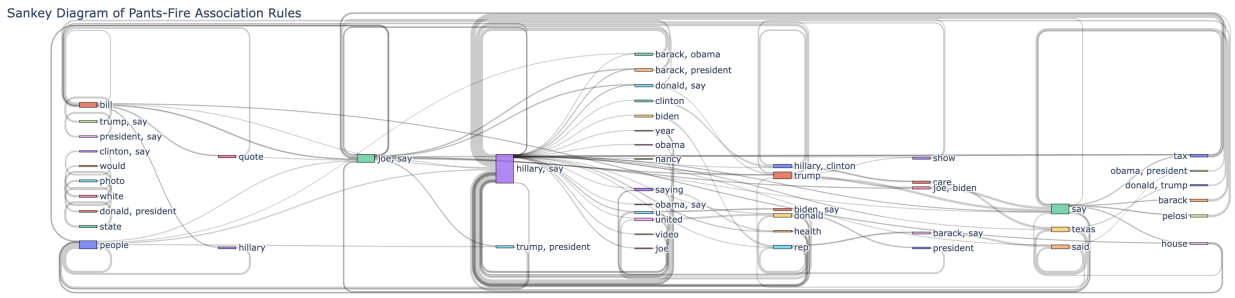


Figure 10.35: Sankey Diagram of Pants-Fire Statement Association Rules

10.3 Appendix C: Glossary of Commonly Used Terms and Acronyms

Application Programming Interface (API): Unlike a user interface, a software intermediary that allows a user to communicate with and retrieve information from a web application.

Artificial Intelligence (AI): Intelligent machines or software that can require training on large amounts of data. Recent versions of generative AI like OpenAI's ChatGPT can imitate chatbots and generate new data based on previous training data.

Bidirectional Encoder Representations from Transformers (BERT): One of the first LLMs developed to expand upon the transformer model framework and is an open-source model used for language prediction and other self-supervised NLP tasks.

Disinformation: Misleading or false information with intent to deceive. In the context of this thesis, opinionated news is an example of disinformation.

Distilled BERT (DistilBERT): An LLM that retains almost the same effectiveness as BERT, however it is lighter to deploy and use.

Large Language Model (LLM): A language model trained on vast amounts of textual data to perform general language generation and prediction tasks.

Latent Dirichlet Allocation (LDA): An unsupervised learning model that is used to classify text in a document or corpus of text to particular topics.

Misinformation: Misleading or false information without intent to deceive.

Natural Language Processing (NLP): Computer programming that converts textual data into machine interpretations of human language.

Pathways Language Model (PaLM): The model framework and API version of Google Bard.

Robustly Optimized BERT Approach (RoBERTa): An LLM developed to outperform BERT through being trained on more data and using a more robust training approach of dynamic masking.

t-Distributed Stochastic Neighbor Embedding (t-SNE): A type of unsupervised learning model that can visualize high-dimensional data clusters using reduced dimensions.

Term Frequency-Inverse Document Frequency (TF-IDF): A measure of importance of a word to a document in a corpus given that words can appear more frequently than others. The TF-IDF formula in the form

$$w_{i,j} = \mathrm{tf}_{i,j} \times \log\left(\frac{N}{\mathrm{df}_i}\right)$$

is used to calculate the weight $w_{i,j}$ of term i in document j, where:

$$\begin{aligned} \mathrm{tf}_{i,j} &= \text{frequency of term } i \text{ in document } j \\ \mathrm{df}_i &= \text{number of documents containing term } i \\ N &= \text{total number of documents in the corpus} \end{aligned}$$

Trustworthiness: A comprehensive measure of misinformation likelihood, of a text, based

on multiple communication factors. A higher trustworthiness means a lower likelihood of misinformation, and vice versa.

Unsupervised Learning: A type of machine learning that does not require training of a labeled dataset to make predictions. An example of this type of learning is generating clusters of data points in a dataset.