# Incorporating Nuance, Complexity, and Interpretability to Misinformation Detection

*Roksana Goworek*


4th Year Project Report
Artificial Intelligence
School of Informatics
University of Edinburgh

2023

# Abstract

This project attempts to increase the nuance, complexity and interpretability of models used for misinformation detection.

This dissertation expands the task of misinformation detection to consider the class of satire by introducing a new dataset, which is then used to compare the performance of logistic regression and BERT, a large language model. Using subsets of articles with distinct topics created through Latent Dirichlet Allocation (LDA), the patterns in classification errors made by both models are analysed to show that BERT is less reliant on the arbitrary statistical properties of the training data than logistic regression. Four probing experiments (of sarcasm, clickbait, sentiment and verb tense) are conducted to investigate the hypothesis that BERT learns linguistic features related to misinformation detection, which could be used for creating an interpretation of classification decisions in terms of human-understandable features. The results support this hypothesis as the model fine-tuned for misinformation detection achieved consistently, but not statistically significantly, better results than the default pre-trained BERT model and BERT fine-tuned on data with randomly shuffled data control models.

This project contributes to the ongoing efforts to combat the spread of misinformation on social media while preserving the principles of free speech and creativity. It attempts to lay the foundation for further research into creating interpretations for the decisions of textual classifiers through probes of high-level linguistic features.

# Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

# Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Roksana Goworek*)

# Acknowledgements

Firstly, I would like to thank my supervisor, Björn Ross, for his support, encouragement and reassurance throughout the entire process. I am also deeply grateful to to Xue Li, Björn's postdoc student, for her invaluable feedback, assistance, and willingness to help at any time throughout the course of my research. I would also like to thank my personal tutor, James Garforth, who has helped tremendously in figuring out my path following graduation, and never turned down a panicked request for a discussion about entering the big wide world of AI.

I would like to extend my heartfelt thanks to all my friends and loved ones, those whom I've known for years to those met in my final year, who have supported me throughout my university studies and the completion of this dissertation. Your unwavering support, encouragement, jokes and tea nights have been a constant source of motivation, and I am grateful for every moment spent with you.

I would like to thank my mum for supporting me and always being there for me when I needed her, and my dad for listening to me ramble incoherently on the bus home from Appleton Tower whenever something went wrong. I cannot conclude without thanking my grandma, who was always there, despite my rare calls, and provided a daily supply thoughtful messages and memes.

Lastly, I would like to thank all the amazing people at this university who have inspired me to be better than I was yesterday every day. Your hard work, dedication, and passion for your respective fields have motivated me to pursue excellence, and I am grateful for the opportunity to learn from and work alongside you.

# Table of Contents

# Chapter 1

# Introduction

## 1.1 Motivation

This chapter presents the motivation behind each part of this project, namely, nuance, complexity and interpretation. It then summarises the main contributions achieved for each part, and concludes by setting up three hypotheses, all of which are addressed and supported by the conducted experiments.

### 1.1.1 The Importance of Misinformation Detection

The spread of misinformation has become more prevalent, with the increasing influence of social media, and can have serious consequences, including the erosion of trust in institutions [57], amplification of prejudices and biases [29], and the potential to influence public opinion and decision-making [2]. Misinformation can also cause personal harm, for example, by spreading false medical advice [5], harmful conspiracy theories [31], and fraudulent schemes [25]. It can also result in destabilising the economy [21], or damage the public image of individuals, organisations and public figures [55]. Censorship of such media is necessary to prevent the spread of misinformation and protect individuals from harm. However, it must be ensured that the censorship is proportionate and can be clearly explained, to avoid overreaching and unintended consequences.

### 1.1.2 The Importance of Nuance

Censorship of social media content can have a disproportionate impact on the creators of satire and the users' access to the content they want. Satirical content often relies on challenging norms and questioning authority [30], thus models used for misinformation detection may label it as misinformative if they fail to grasp the nuance and humour that unmistakably distinguishes satire. This can stifle creativity, undermine free speech, and limit the diversity of perspectives that are shared online [52]. This project expands the task of misinformation detection to consider the class of satire and demonstrates that incorporating this class does not compromise model performance.

### 1.1.3 The Importance of Complexity

While manual feature selection provides transparent explanations for classification, achieving high performance with this approach is very difficult. Dipto Das et al. [23] have used linguistic features to classify articles into the classes ”fake” and ”satire” and were able to achieve an f1-score of 82.5%, which is significantly lower than more complex methods.

Deep learning models give much better classification accuracy in misinformation detection, compared to classical methods [42], through automatic selection of very complex features based on the training data [14]. Furthermore, BERT-based models have been shown to outperform both the classical machine learning methods and other deep learning models by Pavlick et al. [58].

Models with contextual encoders, such as BERT, produce more complex embeddings of text, which significantly increase classification accuracy. This is because deeper layers of the model more complex information is encoded to minimise training loss. However, this also results in the information becoming increasingly more difficult to interpret.

### 1.1.4 The Importance of Interpretability

Intuitively, when humans decide whether an article is genuine, deceptive or satirical, they rely on many linguistic features to make their judgment. They may use concepts of humour, tone, formality and grammatical correctness, among others, to decide the classification of an article. When asked to explain why they classified the article into one of these classes without referencing the facts, or lies, presented in the article, they might point to these linguistic features to explain their decision. In a similar way, a large pre-trained language model, such as BERT, learns complex contextual text representations that might correlate with human-interpretable linguistic features related to the task at hand.

The use of uninterpretable AI models, which seem like oracles to the technologically uneducated public, can harm the users' access to the content they want. This approach to preventing misinformation spread on social media is also more vulnerable to exploitation by those in power to reinforce their beliefs or suppress other viewpoints. We should strive to make decisions made by any opaque system more interpretable so that incorrect decisions can be understood, appealed and potentially prevented in the future.

## 1.2 Contributions

The main contributions of this dissertation are the following:

- I created a more comprehensive dataset of articles for the task of misinformation detection, which includes the class of satire.
- I used logistic regression and BERT to classify said news articles into the classes of true, fake and satire to compare the models' performance, the patterns in incorrect classifications and their potential for interpretability.

- I probe the BERT model fine-tuned for misinformation detection, as well as the default pre-trained BERT and BERT fine-tuned on data with randomly-shuffled labels, on the probing tasks of sarcasm, clickbait, sentiment and verb tense to investigate whether BERT learns linguistic features intuitively related to the task it is fine-tuned for.

I created a suitable dataset of articles for misinformation detection, which includes satire, by concatenating publicly available datasets and proved that logistic regression is adequate at classifying news articles into the classes of fake, real and satire.

I used BERT, a state-of-the-art model in language classification, to improve upon the accuracy of the statistical logistic regression model.

I then investigated both models for explainability. The logistic regression model offers straight-forward explanations for its classifications and, with the use of LDA as a topic modelling technique, I found that there were significant differences in the distribution of topics within different classes of news articles, and the same patterns are reflected in the classification errors made by both models, although to a different extent[1].

Lastly, I probe BERT for the presence of linguistic features using datasets of headlines curated for sarcasm and clickbait detection tasks, as these correspond to linguistic features intuitively associated with misinformation detection, and using labels of sentiment and verb tense created by GPT-3 [12], a state-of-the-art generative language model, on the original dataset to provide comparison to performance on probes directly related to the task the model was fine-tuned for.

This particularly contributes to model interpretability as, when using a linear classifier for the probe, we can measure how much of each linguistic feature the model detects in the text and present that as proxy explanation of the classification decision. This would be valid if further experiments proved more robustly that deep learning modes encode various human-interpretable linguistic features for classification tasks.

## 1.3 Hypotheses

1. News articles can be classified into genuine, misinformative or satirical with sufficient accuracy using statistical analysis methods, when the class of satire is introduced.
2. Employing BERT, a deep neural pre-trained contextual encoder, for this classification task improves classification performance.
3. Large language models encode information related to the task they were fine-tuned for, some of which correspond to human-interpretable linguistic features,in their neurons. Specifically, I hypothesise that fine-tuned BERT will have slightly higher probe performance on classifying features intuitively useful and related to classifying news articles into true/fake/satire classes, and slightly lower performance on classifying other features. The BERT fine-tuned on data with randomly-shuffled labels should have decreased performance on all probes.

[1] As seen in tables 4.5 and 4.10

# Chapter 2

# Background

This chapter first provides definitions of terms used throughout this dissertation. It then further exemplifies the importance of including the class of satire in automatic misinformation detection, followed by a survey of previous research related to satire detection within the field of misinformation detection and the methods employed by these. Subsequently, this chapter talks about interpretability of complex models, from the potential downside of this, through alternative approaches introduced in the literature, to probing, the main method of investigating potential interpretability of deep textual classifiers considered in this project. After conducting a literature review to determine the most relevant linguistic features for misinformation detection with the class of satire, the chapter concludes by compiling a list of potential linguistic features that can be utilized as probing tasks.

## 2.1 Definitions

In this dissertation the terms ”true”, ”fake” and ”satire” are used to mean the following:

**Definition 1.** True news articles relay the facts with no intention to deceive the user and all presented claims are intended to be factually correct.

**Definition 2.** Fake news are intentionally deceitful, either by presenting factually incorrect claims, or an extremely biased and incomplete perspective of a real event.

**Definition 3.** Satirical news presents factually incorrect claims, which are intended to be understood as factually incorrect, and therefore, comedic.

For the sake of brevity, I use the term ”genuine” news interchangeably with ”true” or ”real” news and the term ”misinformation” interchangeably with ”fake” news.

**Probes**
For investigating contextual embeddings of BERT, I attempt to use 4 linguistic features associated with misinformation detection to various degrees. I, or the sources of datasets of these linguistic features, define these to mean the following:

**Definition 4.** Sarcasm in text is the use of irony, which is defined as phrases expressing

the opposite of their literal meaning, or otherwise subverting expectations, to mock or convey contempt.

**Definition 5.** Clickbait, by the original author of the dataset [53], is defined as catchy headlines intended to trick readers into clicking the article, which usually don't live up to the established expectations.

**Definition 6.** Sensationalism is the use of dramatic and exaggerated language to provoke interest, excitement or outrage.

**Definition 7.** Sentiment of an article is a general attitude or opinion presented by the author, but in this project it is used as a rating of how positive or negative the text is.

## 2.2 Satire

Despite being factually incorrect, satirical articles should not be classified under the same umbrella as fake news as they do not have the same malicious intentions, nor the negative impact of intentionally disingenuous articles. Satirical articles aim to be understood as satire and when they are misinterpreted by some as genuine, these mistakes are quickly corrected by online communities. Satire employs humour, sarcasm, hyperbole and ridicule, among many other linguistic features, to criticise and highlight flaws in people, organisations, regulations and situations, particularly in the context of politics and social issues to entertain their readers.

While some studies claim that much of the news that people engage with on social media is deceitful [70], this might be an exaggeration of the problem, ranging with the strictness of the classification of fake news [63]. Flagging satirical journalists as 'fake' penalises them unfairly and prevents the spread of their content, potentially harming them financially. There are large audiences who enjoy watching or reading satire for entertainment [34], and most people can easily differentiate between satire and genuine news. Even Facebook has agreed that satire should not be classified in the same way as malicious misinformation, and added an option to mark content as "Satire", if the article's source is a known satire publication, or if a reasonable reader would understand it to be irony or humor [27].

While some users might not realise the joke behind the articles [6], the point of satire is for the reader to understand the farce and be in on the joke. Therefore, satirical writers leave many hints exposing the insincerity of their claims, which makes them humorous. The potential negative impact of someone misunderstanding a satirical article and sharing it as fact does not compare to the harm that malicious misinformation can have. In most cases, it will quickly be noticed by other internet users and pointed out as satire, often ridiculing the user who fell for it, as evident in a subreddit r/AteTheOnion, created specifically for this purpose.

One study analysing the use of sarcasm and irony within amazon reviews [28] gauged whether people understood the intended meaning despite the sarcasm by asking people to guess the amazon product rating. They did so with high accuracy (with correlation 0.821 to the actual rating for sarcastic reviews), showing that people are really good at understanding sarcasm, suggesting that people are also not easily fooled by satire.

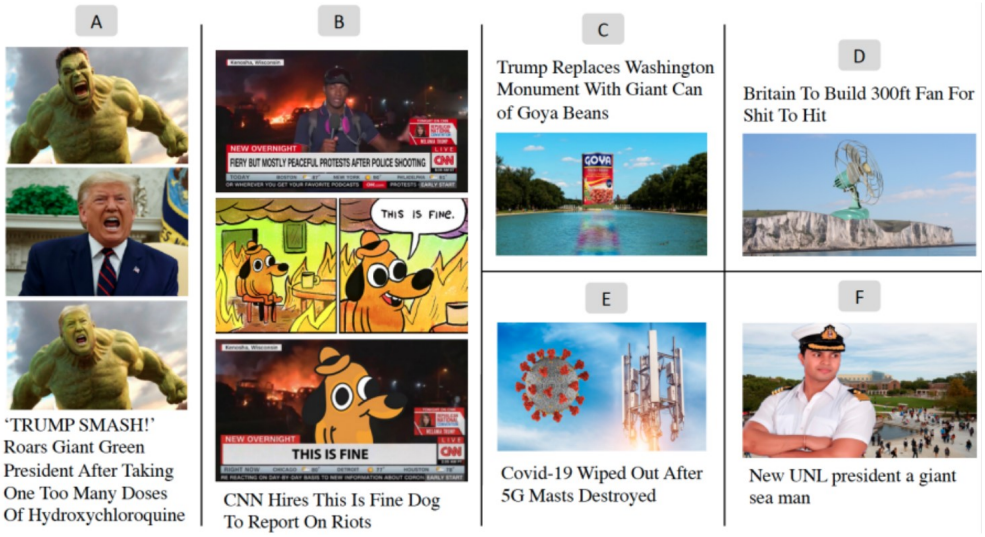



Figure 2.1: A Selection of Satirical News Headlines Collated by Li et al. [49], Exemplifying the Ease of Identifying Satire for Humans.

## 2.3 Complexity

This project uses logistic regression as a simple statistical model to provide baseline results and give insight into trends in the dataset, as well as these classes in general.

Khan et al. [42] conducted a benchmark analysis of the effectiveness of various machine learning techniques in detecting fake news across three distinct datasets. Their findings revealed that pre-trained language models, such as BERT, showed superior performance. Considering this, and the findings summarised below, BERT was employed to get the highest possible classification accuracy on the misinformation detection dataset. A technical explanation of BERT can be found in §3.6.

### 2.3.1 BERT

Most modern approaches to text processing involve using Transformers, particularly Google's BERT (Bidirectional Encoder Representations from Transformers). It has been widely adapted as the state-of-the-art method in text classification [73] and misinformation detection [40]. The deep contextualising nature of BERT makes it perfectly suited to this task, as shown by the plethora of papers proposing BERT-based models for misinformation detection [39] [51] [45]. Its use results in excellent performance, as exemplified by Kaliyar et al. [40], who propose a modification to the traditional BERT architecture to create FakeBERT, which outperforms other models in misinformation detection with an accuracy of 98.90%.

BERT has also been employed in numerous studies in satire detection [48]. It has also shown impressive results in satire detection on French and Romanian datasets:

**FreSaDa: A French Satire Data Set for Cross-Domain Satire Detection** [37] created their own dataset from publicly available websites. They used CamemBERT (French BERT), which achieved a staggering accuracy rate of 97.48%.

**SaRoCo: Detecting Satire in a Novel Romanian Corpus of News Articles** [64] used RoBERT (Romanian BERT) for this classification task. They employed an interesting check of the accuracy of human classification, by asking human test subjects to classify the same headlines as the model and found that the human accuracy was around 87%, while their model achieved 68% accuracy on the same news headlines. On full news articles the model's accuracy increased to 73%.

## 2.4 Interpretability

AI models for misinformation detection are not perfect, they amplify the bias of their training set, and can make incorrect classifications.

### 2.4.1 The Downside of Making AI interpretable

Transparency makes AI models more vulnerable to exploitation, for example, in the way that spammers adapted their language to avoid being detected by automatic spam filters [38]. [68] have shown that providing explanation of a model's decision not only makes it vulnerable to attacks, but also risks exposing the potentially private information included in the training set. However, interpretability is needed to ensure that model decisions are fair and based on features we agree with. If we found that a model learned linguistic features that we deem unethical to be a part of the decision process, we should take steps towards reducing the uncovered bias.

### 2.4.2 Interpreting BERT

The current methods of interpreting large Transformer-based language models rely on backtracking attention to visualise which parts of the input they pay most attention to.

Bracsoveanu et al. [11] conducted a survey of different visualisation methods for Transformers. From visualising neuronal nets to attention maps, they analyse which parts of the encoder are considered in the visualisation and give a good overview of the best approaches to interpreting Transformers. The tools they look at include:

- BertViz [76]
- Clark [20]
- VisBERT [1]
- ExBERT [35]
- AttViz [69]
- Kobbayashi [44]

Additionally, Chefer et al. [18] use vision Transformers and highlight which parts of the question, as well as which parts of the image about which the question is asked, are paid most attention to.

Riberio et al. [62] have produced a tool, LIME, which explains the classification decision of classifiers by highlighting which parts of the input text negatively or positively influence the probability of the text belonging to a certain class, similar to the interpretability offered by logistic regression. LIME was evaluated by Szczepanski et al. [72] for the task of classifying headlines into real or fake.

Hao et al. [32] attempt to produce interpretation for BERT from its multiple attention heads and take into account how input features interact with each other, not only their salience for each class. For each layer they extract the most salient feature interactions to create a salience tree, revealing the hierarchical interactions inside the Transformer. This form of explanation is extensive and takes into account the complexity of BERT embeddings in different layers and attention heads, however, it might not be interpretable to those who are not familiar with linguistics or neural networks.

All of these methods offer different advantages and have different shortcomings, but they are not comprehensive, or not understandable to people who are not familiar with Transformers or neural networks in general.

Kim et al. [43] have introduced Concept Activation Vectors (CAVs) for visual neural network classifiers, which explain classification decisions in terms of human-friendly high-level concepts. Little to no similar work has been done for Transformers or textual neural classifiers as high-level textual features cannot be easily extracted or understood from vector embeddings, which this dissertation attempts to preliminarily address through probing.

## 2.5 Probing

Visual deep learning classifiers have long been known to learn high-level features related to the classification task [46]. I hypothesise, as stated in §1.3, that deep neural text classifiers encode high-level linguistic information in a similar way, but it is difficult to identify which (if any) features they correspond to because linguistic features cannot easily be averaged over. One method for investigating this hypothesis is through the use of probes. A technical explanation of probes and probing tasks is given in §3.7.

Below is a brief presentation of the relevant research on probing tasks, which influences a lot of the decisions taken in constructing probing tasks for my experiments.

**Semantic Information** - Tenney et al. [74] investigated whether contextualised word representations, including BERT, encode semantic or syntactic information more, and compared these to classical statistical methods of word encoding. They found that, although contextual word representations yield improvement for both semantic and syntactic tasks over statistical methods, surprisingly, the improvement on semantic tasks is smaller. In the probing experiments for this project, probing tasks are used to investigate various semantic phenomena such as sarcasm, sentiment, and clickbait, as well as verb tense, which is more closely related to syntax than semantics. Tenney et al.'s results suggest that contextualised embeddings would perform better on these probes than statistical embeddings.

**Random probes** have also been shown to have good performance on probing tasks

by Zang et al. [80]. When all weights above the lexical layer, layer 0, were replaced with random orthonormal matrices, the performance of that neural network was still comparable to trained models. This inspired the approach to use BERT base with weights updated randomly, introduced in §3.6.3, to the same extent as the model fine-tuned for misinformation detection, as a control baseline for comparison of probing task results. This was done to ensure sufficient comparison of the effects of fine-tuning contextual encoders, even if pre-trained BERT had better probe performance than fine-tuned BERT as it has more general linguistic knowledge.

**Top Layers** - Some studies of contextual embeddings, including Peters et al. [59], Blevins et al. [10] and Tenney et al. [74], have found that top layers might contain very specialised knowledge and hence the input of intermediate levels might have better performance on typical probing tasks in NLP. This further supports the decision to create specialised probing tasks of linguistic features related to the task being fine-tuned for and to investigate the final-layer embeddings for these tasks.

### 2.5.1 Simplicity of Probing Tasks

According to Conneau et al. [22], probing tasks are simple in order to minimize interpretability problems. The extended use of simple probing tasks could also be partially explained by the availability of reliable labelled datasets, or the assumption that these tasks are representative of a model's overall linguistic abilities and understanding, which allows for comparison between models and tasks. However, Alt et al. [4] found that models with self-attention, while achieving state-of-the-art performance on the main relation extraction task for which the models were trained for, have lower probing task performance than other encoding architectures. They suggest that this could be due to the fact that self attention encodes deeper linguistic information, which is not covered by their probing tasks, further motivating the decision to employ probes of complex linguistic features related to the task the model was fine-tuned for.

Hence, linguistically complex probing tasks related to misinformation detection in varying degrees were used for experiments within this dissertation to test whether performance is higher on tasks semantically related to the task the model was fine-tuned for.

### 2.5.2 Complexity of Probes

The complexity of the probe, i.e. the classifier used on top of a neural network to investigate its embeddings, significantly affects what can be learned from a probe's performance. A more complex classifier could simply be learning the features regardless of the information explicitly encoded in the model's embeddings, as explained by Hewitt et al. [33]. They highlight the importance of reducing the amount of training data and using smaller classifiers for probes.

The classifier should have limited expressive power on its own so that its performance helps to isolate the contribution of the embeddings towards task performance. A complex classifier, such as a multi-layer neural network can apply non-linear transformations to the inputs, allowing the model to extract different features than what is directly encoded

in the embeddings. Additionally, a simple classifier would be easier to interpret and analyse in the future, allowing us to understand which embeddings are most important for which probing task. For these reasons, the linear classifier that is by default on top of BERT is used as the probe for all my experiments.

## 2.6 Linguistic Features

### 2.6.1 Features Typical of Satire

Satire aims to expose its own falsehood in a way that is humorous and exaggerated, and there are many linguistic features that alert readers to the author's true intentions. Different papers have identified different features that are characteristic of satire. For instance, Burfoot and Baldwin [15] identified profanity, slang, and headlines as typical features of satire, while Li et al. [49] found heavily-edited and over-dramatized thumbnails to be indicative of satirical articles. Das et al. [23] investigated the tone of satirical articles, looking specifically at the language, emotion, and social scores, while Yang et al. [79] examined psycholinguistic, stylistic, readability, and structural features.

### 2.6.2 Linguistic Features Useful for Classification of News Articles

Burgoon et al. [16] and Zhou et al. [81] found that the complexity of language used in fake news is lower or different from the complexity of language used in genuine news articles. Rubin et al. [65] identified specific humor, grammar, negative affect, absurdity, and punctuation as features that distinguish satire from other types of news. Levi et al. [48] found that satirical articles are often written in a more formal tone than fake news, and that satire is often humorous and political.

### 2.6.3 Linguistic Features for Probing Tasks

I hypothesize that BERT fine-tuned for misinformation detection would have higher accuracy on probing tasks constructed from datasets of the above features than a default pre-trained BERT, or a model fine-tune for another task. The above findings were used to compile a list of linguistic features informative for misinformation detection, ranked in order of how much one would expect them to be aligned with the classification task:

- Sarcasm
- Absurdity
- Presence of political themes
- Sensationalism
- Sentiment
- Formality
- Complexity
- Verb Tense

The presence of sarcasm and absurdity are most indicative of satire. The presence of political topics, although not a linguistic concept in itself, seems to be highly aligned with the classification task. Sensationalism is often indicative of unreliable news. The sentiment of the article has been shown to be correlated with the reliability of the news article [3]. The formality, complexity and verb tense of articles are not intuitively informative for this classification task, hence they could be used as control linguistic features not intuitively related to misinformation detection to ensure that a fine-tuned model does not simply have higher performance on all probing tasks than the default pre-trained and randomly-trained control models.

Sarcasm, sentiment, verb tense, and clickbait, were used as probing tasks to compare the performance of BERT fine-tuned for misinformation detection with the performance of BERT fine-tuned on data with randomly shuffled labels and a default pre-trained BERT.

# Chapter 3

# Methods

This chapter presents the technical details of methods used in this dissertation. These include descriptions of the datasets used to create the main dataset containing the classes of true, fake and satirical news articles, and the pre-processing steps applied to this dataset to ensure that the models could achieve the highest possible performance. It then explains LDA, which was used to cluster articles into topics for future analysis, and word clouds, which are used to visualise these topics. Then, the two models, logistic regression and BERT, are explained in technical detail, along with the types of embeddings that are used or created for each model and the potential methods of interpreting their classification decisions. This leads into a technical explanation of probing tasks, the considerations that must be taken in creating and interpreting these, as well as an overview of the linguistic features selected for probing tasks.

## 3.1 Datasets

For the purpose of creating a balanced dataset of true, fake and satirical news articles, multiple datasets were collated. All of the used datasets contain news articles written between 2015 and 2017. The single source of satirical articles was the dataset created by Fan et al. [79], which contains over 16 thousand satirical articles and 160 thousand genuine news articles. One limitation of the satire dataset is that it only included the body of articles. Although this limited the scope of experiments, as metadata can be important for classification, the dataset was sufficient for all experiments, which focused on classification and interpreting the model's decision based on the language used in input articles. There were very few other datasets that included satirical news, and those that were publicly available contained less than 300 articles.

Two additional datasets were included to introduce the class of 'Fake' news and add variety of sources to the dataset.

The "Fake and Real News Dataset" [9] consists of two files, one with 23,481 fake and one with 21,417 genuine articles. Each entry contained the title, text, subject (news/politics/other) and date of the article.

The kaggle community competition dataset [50] consisted of three files: train.csv, with

each entry containing the id, title, author, text and label (reliable/unreliable) of the article, test.csv, with the same attributes as train.csv, except without the label and submit.csv, which contained the ids and labels of the test.csv entries.

All of the datasets were combined into a complete dataset containing only the text, label (fake/true/satire) and source of each article.

The distribution of classes from the different sources is as follows:

| Dataset | True | Fake | Satire |
|---|---|---|---|
| Fake and Real News | 21,417 | 23,481 | 0 |
| Kaggle Competition | 12,726 | 13,228 | 0 |
| Satire dataset | 168,780 | 0 | 16,249 |

Table 3.1: Distribution of Classes from Different Datasets

## 3.2 Pre-processing

To extract the most meaningful information from the articles, the following pre-processing steps were applied to the dataset for both models:

- Removal of empty entries and duplicates
- Removal of all-capital leading words longer than one character, as some articles from the ”Fake and Real News” dataset started with a word or phrase denoting the location of the event in block capitals
- Removal of html tags and links
- Removal of unnecessary whitespace
- Removal of punctuation at the beginning of text

The dataset was then duplicated into a BERT dataset (dataset to be used with the BERT model), which required no further pre-processing as BERT is a large pre-trained language model that is designed to handle real language, including stopwords and punctuation, and a logistic regression dataset.

The following steps were applied to the logistic regression dataset:

- Tokenization
- Expansion of contractions (’isn’t’ $\rightarrow$ ’is not’)
- Punctuation removal
- Stop word removal
- Lemmatization (’running’ $\rightarrow$ ’run’)
- Swapping of all numbers for a ’num’ tag
- Lowercasing

Afterwards, the logistic regression dataset was filtered for any empty or duplicate entries that may have been created during pre-processing. These were removed from both datasets.

### 3.2.1 Training, validation and test split

To evaluate the performance of models, the dataset was split into training, validation, and test sets in the ratio 80:10:10. The training set was used to fit the model and the test set was used to evaluate the final performance of the model. Although the validation set wasn't used for hyperparameter tuning due to the high accuracy of models with default parameters, a validation set was created to evaluate the model during the fine-tuning process. This allowed for the possibility of addressing any issues such as overfitting, and make changes to the dataset or models before a final comparison of model performance.

## 3.3 Latent Dirichlet Allocation (LDA)

To gain insight into the reasons for incorrect predictions and explore whether certain topics are more likely to be misclassified than others, LDA was used as a topic modeling technique.

By seeing how models are likely to make mistakes, we can improve their performance on the data that is most likely to be misclassified. This approach to general misinformation detection relies on the assumption that the dataset is representative of news articles in general. The dataset created and used for this project is likely to be representative of many different news articles as it is large and originates from three different sources, although, these sources were all crafted for the task of misinformation classification, so they might have some specific traits that are not representative of all news articles in general. Based on the information available, they are sufficiently representative to effectively serve as a proof of concept for this project.

LDA, introduced by Mimno et al. [56] is a probabilistic model that represents each article as a mixture of a fixed number of topics, and each topic is represented as a distribution over the words in the vocabulary. It makes the assumption that the articles are generated from a mixture of these latent topics, and that each topic is a collection of words that occur together more often than would be expected by chance. The goal of LDA is to uncover the underlying topics in the articles, allowing us to further understand patterns in misclassifications made by different models and what could be causing them.

Although LDA is not typically a clustering algorithm, as it produces a distribution of topics over articles, we can approximate clusters by assigning the highest-probability topic to each article. An alternative approach to analyzing the topics of misclassified articles, while leveraging the probabilistic nature of LDA, would be to use the distribution of topics of articles directly. For example, when examining which articles were misclassified, we could investigate the distribution of misclassified vs. correctly classified articles in a continuous multi-dimensional space corresponding to the distribution of topics. One could also investigate the mean distribution of topics of articles misclassified in the same way (e.g. satirical articles misclassified as false). However,

interpreting the results of this approach can be challenging, as a wide range of topics will be misclassified in each possible way, and the mean of their distributions may not be very informative. Further research into this method, such as using k-means clustering to group topic distributions into 'types of articles', could be interesting, but is beyond the scope of this dissertation.

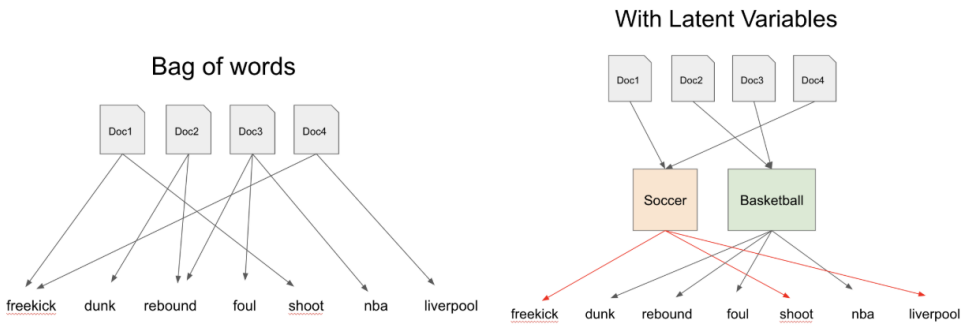



Figure 3.1: Visual Explanation of Latent Dirichlet Allocation. Figure sourced from [67]

The implementation of LDA for topic clustering was based on this tutorial [60]. To determine the number of topics for clustering the dataset, an analysis was conducted on the topic coherence using approaches similar to those presented in [41] and [54], but ultimately, the number of topics was selected based on the ease of labelling them based on their top 10 highest-weighted words, as their purpose is interpretability of misclassifications.

## 3.4 Word clouds

A word cloud is a visual representation of the frequency of words in a corpus, in which more frequent words are displayed in a larger font and less frequent words are displayed in a smaller font. By comparing the word clouds of words typically found in misclassified articles to those of correctly classified articles, we can determine whether misclassified articles have language characteristics that are more similar to their predicted class rather than their true class. This can provide insights into the factors that may be contributing to misclassifications.

## 3.5 Logistic Regression

### 3.5.1 (Multinomial) Logistic Regression

Multinomial logistic regression was used to classify embeddings (explained below in §3.5.2) of news articles in the dataset into one of three classes: fake, real, or satire. Logistic regression works by predicting the probability of the input belonging to each class for a given input using the logistic function,

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

and assigning the input to the class with the highest probability.

After fitting of logistic regression weights, a column of predictions was generated by the model on the dataset for further analysis of misclassified articles. The implementation of the logistic regression model was based on this tutorial [13].

#### 3.5.1.1 Random Logistic Regression

For a baseline measure of how much logistic regression learns in comparison to a random model, another model was fitted to training data with randomly-shuffled labels.

### 3.5.2 TF-IDF embeddings

Tf-idf embeddings were used in order to represent the articles in vectors suitable for logistic regression. Tf-idf (term frequency-inverse document frequency) is a method for representing the importance of a word within a document in relation to a corpus of documents. It allows us to weight the words in the articles, so that words that are more important for understanding the content of the article are given higher weights.

The term frequency, $tf(t,d)$, is the frequency of term t in the document d, shown in the equation (3.1).

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \tag{3.1}$$

The inverse document frequency, $idf(t,D)$, is the logarithmically scaled inverse fraction of documents that contain the term, and intuitively measures how common the term is across all documents:

$$idf(t,D) = log\frac{N}{|\{d \in D : t \in d\}|} \tag{3.2}$$

Where N is the number of documents in the dataset, $|D|$.

The term frequency-inverse document frequency is the dot product of these two.

$$tfidf(t,d,D) = tf(t,d)\dot{i}df(t,D) \tag{3.3}$$

The tf-idf values of the words in each article were used as input features for the logistic regression model, which was then able to predict the class of the article based on those features.

These statistical embeddings are used as a benchmark and point of contrast against BERT's contextual embeddings. Tf-idf embeddings were used instead of word2vec or other word embeddings because they are better at capturing the meaning of individual words in relation to the documents they appear in, making them more suitable for classification tasks. Additionally, tf-idf embeddings are simpler to compute and interpret, and require less computational resources to generate, making them a good choice for the large dataset and allowing for simple and fast interpretability of misclassified examples.

### 3.5.3 Interpreting Logistic Regression Results

One of the key advantages of logistic regression is that it is interpretable. The coefficients for each input feature can be used to understand how much each feature contributes to the predicted class probabilities. In the experiments, the features take the form of tf-idf word embeddings, which can be traced back to the original words using the mapping created when calculating the tf-idf scores. For example, if the logistic regression coefficient for a particular embedding is large and positive for a particular class, it means that an increase in the frequency of the word represented by that embedding in the document is associated with an increase in the probability of that document belonging to that class. And vice versa.

## 3.6 BERT

BERT was used to achieve better performance on classifying articles in this dataset. Pre-trained BERT (Bidirectional Encoder Representations from Transformers) [24] is a language model based on a deep Transformer architecture [75], specifically the encoder, trained jointly as a masked language model and on next-sentence prediction using a concatenation of the Toronto Books corpus (Zhu et al., 2015 [82]) and English Wikipedia.

The Transformer [75] encoder reads all the words in the input sequence at the same time, rather than reading the text input in a linear or sequential way, allowing BERT to better understand the meaning of the whole input text and how different parts of it relate to each other to form the overall meaning.

It is designed to understand the contextual relationships between words by creating powerful embeddings that can be used in various text classification tasks, such as misinformation detection.

### 3.6.1 BERT Embeddings

The tf-idf embeddings employed for the logistic regression model are static in regards to each word, representing the importance of that word for each article. The tf-idf embedding of a given word is going to be the same, regardless of where in the corpus it appears. On the contrary, BERT represents words based on their context in the text. That way, the word ”run” has different representations for ”run a business” and ”run a marathon” due to their context.

First, BERT employs a tokenization technique called WordPiece [66], which involves breaking words into individual tokens or smaller ”word pieces”. For example, a single word might be represented by one token or it could be broken down into multiple tokens [24].

This greatly improves upon the manual pre-processing of the dataset, through tokenisation, lemmatisation and stemming for the logistic regression model, as it splits up words into their component pieces and maintains all semantic knowledge conveyed by


Figure 3.2: Example of WordPiece Tokenisation, taken from [71].

all parts. This way, many long unknown words can now be recognised by the model, even if they were not seen in the training data.

### 3.6.2 BERT: Encoder of a Transformer

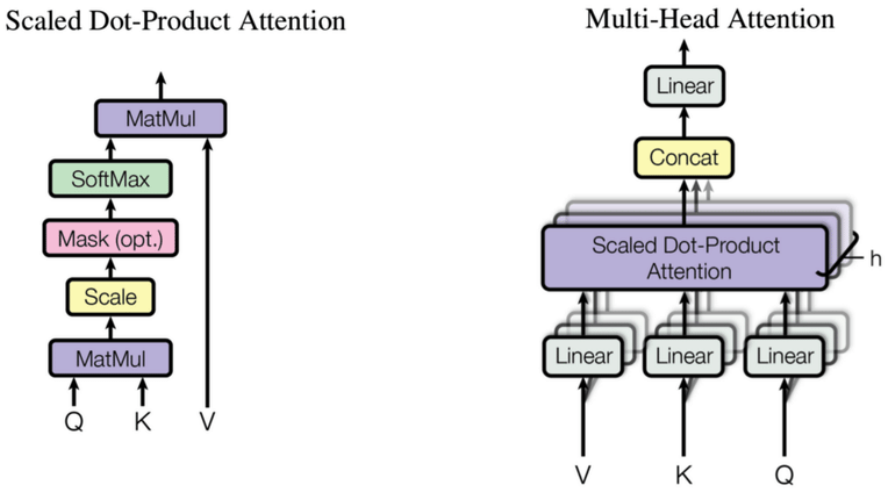



Figure 3.3: Computation of Multi-Head Attention from Vaswani et al. [75]

BERT consists of the encoder part of a Transformer (which is made up of an encoder and decoder). This encoder consists of two parts: an attention layer and a feed-forward network. The self attention layer calculates how much attention should be paid to each part of the input to produce each part the output. To do this, the encoder uses the trained key, query and value matrices to calculate an attention score for each input vector. It does this by taking the dot product of that input vector's query vector and the key vectors of all other input vectors. The model determines how much it should attend to each input vector by applying softmax to the results of these dot products. This gives the attention score for that input vector, which prescribes how much attention the model should pay to each input vector to compute the contextual embedding for this input vector. To calculate the encoding of a vector, each input vector is multiplied by its attention score corresponding to the current vector, and the scored vectors are summed

together.

BERT base consists of 12 layers of self-attention, with 12 attention heads per layer. Multi-head attention works by learning separate key, query and value matrices for each attention head. These learn to attend to different information as they are trained to minimise the same loss function. The scaled dot product attention from all attention heads are concatenated and passed through a linear transformation to produce multi-head attention.

The explanation of how the BERT model was fine-tuned can be found in Appendix C.

### 3.6.3 Random BERT

Another model was trained in the same exact way, with the same hyperparameters and data, except the target labels were randomly shuffled, in the same way as for the randomly-fitted logistic regression model introduced in §3.5.1.1. This was done to:

1. Randomly change BERT weights to the same extent as the fine-tuned BERT to have a comparison of how this would compromise BERT's performance on probing tasks.
2. Compare how BERT and logistic regression deal with uninformative training data.

### 3.6.4 Investigating the Misclassified Articles

I expect BERT text embeddings to contain more semantic information than tf-idf word embeddings due to their contextualised nature. This helps to model the semantic meaning of the input text, which is crucial for classification and may allow BERT to find more interesting patterns. BERT embeddings are based on the very large dataset (3.3 billion word corpus), as well as the dataset of articles used for fine-tuning, while tf-idf embeddings are based only on the frequency of words in each article in the dataset. The pre-trained nature of BERT makes its embeddings more generalisable to unseen articles.

To understand the behavior of the model and identify any issues with its predictions, a new column was created in the dataset containing the class labels predicted by BERT. The same was created for predictions made by the logistic regression model. Then a confusion matrix from the predicted labels on the unseen (validation and test set) articles was created to analyze the misclassified articles and compare the predictions made by BERT to those made by the logistic regression model. The distribution of these misclassified articles across the four topics, as identified by LDA, was analysed in order to understand what may be causing these errors.

### 3.6.5 Interpreting BERT

In order to take a step towards human-interpretable explanations of the decisions made by neural text classifiers, such as BERT, probing tasks were used to determine

whether embeddings created by BERT fine-tuned for misinformation detection contain information that correlates with linguistic features intuitively related to this task.

Although this does not directly provide interpretation for BERT's classifications, it paves the path forward to making text-processing NNs more interpretable even for people with no understanding of neural networks. If linguistic features related to the main task are approximately encoded in the final-layer embeddings of a model trained for that task, then we can use the outputs from the linear probes to get measures of the presence of linguistic features corresponding to these probes as identified by the model. Thus, we can see how much of each linguistic features in the set the model "sees" in a new unseen article, and present those values, paired with the label of that linguistic feature, as proxy explanation of some of the model's considerations in the making the decision. This relies on having good datasets of linguistic features that can reliably show the correlation with said features in a model through probing.

Unlike the attention-based visualisations and explanations of model decisions, these explanations would not be exhaustive as the combinations and complexities of neural networks would likely not be exhaustively covered by all the linguistic features considered, but they might give insights into what complex linguistic patterns are discerned by the models.

## 3.7 Probing

A probing task is a task on detecting or classifying a (linguistic) feature of interest. A probe is a simple classifier trained for this task using embeddings created by the contextual encoder being investigated, such as BERT. They are used to see whether BERT encodings contain linguistic information related to the task it is fine-tuned for. These could, in turn, be used to generate measures of the presence of linguistic features for each input that BERT 'looks at', but this is recommend for future work and is beyond the scope of this dissertation.

### 3.7.1 Acquiring Labels

As there is a lack of datasets and reliable models of the linguistic features that are intuitively related to this nuanced misinformation detection task, OpenAI's API was used to label the existing dataset with ratings and labels for some of these features to serve as probing tasks. To maintain accuracy of predictions and ensure that GPT-3 (text-davinci-03) follows the prescribed output format, the list of features to be investigated was limited to:

- Sarcasm
- Sensationalism
- Sentiment
- Verb tense

as these represent a range of association with misinformation detection.

The current state-of-the-art language generation models, text-davinci-003 by OpenAI [26], was used to label articles with these linguistic features.

The temperature parameter in calling the GPT-3 model intuitively corresponds with the scale from high factual correctness to highly imaginative responses. The value of 0.3 was selected by performing a grid search of values from 0.1 to 0.9 in 0.1 increments, and manually inspecting responses from the model to ensure they align with human judgements of these articles, are factually correct and in the requested format.

The following prompt was used: "*Is the below article sarcastic? Is the article sensationalised? Is the article mostly positive, negative or neutral? What verb tense is it mostly in? Reply with only a "yes" or "no" for sarcasm, a "yes" or "no" for sensationalism, the sentiment of the article and tense, e.g. "yes, yes, positive, past tense". Here is the article:*"

GPT-3 responses are limited to 50 tokens to compel it to generate responses in the requested format. The output is saved by splitting the list generated by text-davinci-03 into a new column in a csv file, along with the beginning of the article text it corresponded to in order to validate that it was matched with the correct article when merging with the main dataset.

Due to the financial cost of querying GPT-3, only 10% of the dataset was selected randomly to be labelled. After postprocessing the results and removing 1,113 articles due to nonsensical answers, the resulting dataset, to be used for training and testing the linear probes, contained 23,430 articles with labels for satire, sensationalism, sentiment and verb tense, and the 80-10-10 training, validation and test set ratio was preserved.

Although probing tasks are normally taken from different datasets to avoid discovering trivial correlation with the original trained-on target labels, two of the probing tasks for this project (sentiment and verb tense) were created by labelling the original dataset. This was done to mimic what could be done in future research to make neural text classification explainable in terms of many different linguistic features, even if there is a lack of high-quality publicly available datasets for all of them.

### 3.7.2 Unreliability of GPT-3 in Creating Labels

Using a large language model (LLM) to label the dataset is dependant on the accuracy of the LLM being used. This method poses the risk of unreliable probing datasets, which would partially invalidate results of probing tasks. However, performance of LLMs on NLP tasks in recent years has increased exponentially. GPT-3 [26] has been shown to have excellent performance on all standard NLP tasks [47], and its application in labelling tasks has already been explored by Wang et al. [78]. Along with manual inspection of select labels and parameter tuning based on alignment with human intuition, this was not a large concern for this model.

#### Rejecting Sarcasm and Sensationalism

Unfortunately, due to the very low amounts of articles being classified as sarcastic (19) and sensationalised (1), these did not make suitable datasets for probes and hence these probes were replaced by publicly available datasets of sarcastic and clickbait titles.

A further explanation and exploration of these articles can be found in Appendix A. Instead, publicly available sarcasm and clickbait datasets, described below, are used for additional probing tasks with linguistic features related to the misinformation detection task.

### 3.7.3 Sarcasm

The sarcasm dataset for the probing task was obtained from the huggingface website [61]. This dataset contains headline, article link and sarcasm label for each article. They collected sarcastic headlines from The Onion and non-sarcastic headlines from HuffPost. These sources might risk overlapping with the sources in the dataset created for this project, however, since only headlines are used in this dataset and only article text is used in the dataset used for this project, there is no direct overlap of training data and probe dataset.

### 3.7.4 Clickbait

The clickbait dataset for this probing task was also obtained from the huggingface website [53]. This dataset contains the titles of clickbait and non-clickbait articles, as well as the label for whether it is clickbait or not. The dataset was introduced by Chakraborty et al. [17] for detecting and preventing clickbait online. Similarly, since only titles of articles are used, there is no risk of overlap with the dataset of article text.

### 3.7.5 Training Probes

Subsequently, for each BERT model, the BERT fine-tuned for misinformation detection, base BERT and randomly fine-tuned BERT, I created four probes, one for each of the linguistic features being investigated.

#### 3.7.5.1 Linear Classifier

The final layers of the BERT model, following the 12 blocks of attention are:

```
bert.pooler.dense.weight
bert.pooler.dense.bias
classifier.weight
classifier.bias
```

The final classifier layers (one for weights, one for the bias) in bert-base-uncased consists of a linear transformation followed by a softmax activation function shown in 3.4 to obtain class probabilities, outputting the highest-probability class.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.4}$$

for $i = 1, ..., K$ and $z = (z_1, ..., z_K) \in R^K$, where z is the input vector and K is the number of classes.

The linear transformation ensures that information encoded in embeddings is preserved, which is why this is appropriate for investigating the presence of linguistic features in these embeddings. The probe (classifier) is the only part of the BERT architecture that is trained on the probing data. Fine-tuning this linear classifier without altering the parameters of the model required freezing all the attention layers and using the huggingface trainer API [36].

# Chapter 4

# Results

This chapter explains how this project accomplished all of the goals set up in the introduction §1.3. I set out to create a more nuanced and comprehensive dataset of news articles for misinformation detection, which included the class of satire. I intended to use logistic regression and BERT for classification of these articles and compare their performance, as well as the patterns in their classification errors in terms of human-understandable topics. I then intended to investigate the hypothesis that BERT learns linguistic features intuitively related to the task it is fine-tuned for through probing tasks related to misinformation detection in varying degrees. This chapter begins by presenting various article distributions and patterns in the dataset, followed by the results of logistic regression and BERT models. It concludes by presenting the results of the three BERT models on the four probing tasks and explains why the results are not merely reflecting the model's ability to perform the task it was fine-tuned for.

## 4.1 Overall Statistics and Additional Information

To put results into perspective and make it easier to analyse any out-of proportion predictions, here is the distribution of classes in the whole dataset, which is upheld within $\pm 0.3\%$ in all subsets. To analyse most of the results, a separate binary F1-score,

| Class | Proportion |
|---|---|
| True | 81.20% |
| Fake | 12.19% |
| Satire | 6.61% |

Table 4.1: Proportion of classes across the whole dataset

which is a harmonic mean of the precision and recall of predictions, is used for each class in order to compare the models' performance on different classes.

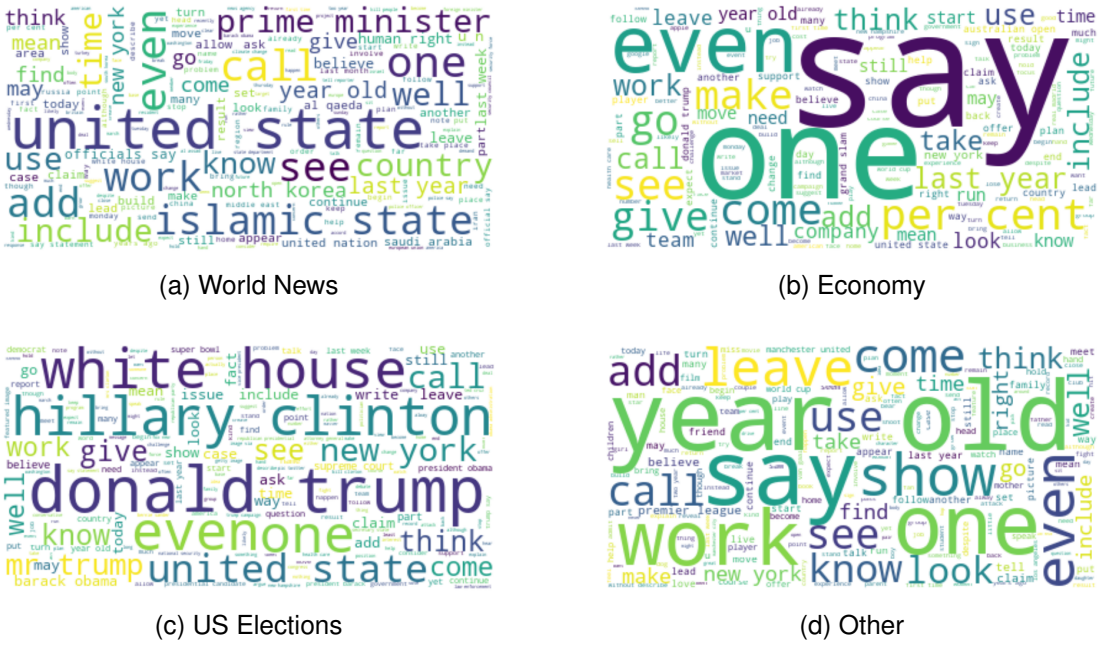
The F1-score formula is as follows:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{4.1}$$

## 4.2 LDA Topics

As described in §3.3, LDA was used to automatically cluster articles into four topics. The performance on articles in separate topics was analysed to understand whether there are semantic patterns in the way BERT or logistic regression classify articles. Since the latter is purely based on statistical differences, its classification can only be explained by imbalances in the training data.

Only 4 topics were selected as higher numbers of topics resulted in less intuitively-coherent topics, based on the 10 highest-weighted words per topic of the LDA model. As each article was represented as a distribution of topics, each article was labelled with the topic that had the highest weight associated with it to create discrete subsets of topics. The resulting subsets of articles were used to produce the below WordClouds, introduced in §3.4.




(a) World News (b) Economy

(c) US Elections (d) Other

Figure 4.1: WordClouds of LDA Topics

As can be seen in 4.2, the distribution of topics is relatively even across the whole dataset, with the proportion of articles within each class ranging between 13% and 31%. This is mostly preserved in the True articles, as the proportion of topics ranges between 15% and 34%. On the contrary, the classes of satire and fake news have more extreme distributions of topics, with large majorities at 58.03% and 66.61% respectively, and small minority classes, with 6.90% and 5.32% respectively. A lot of the Fake articles are clustered in the 'US Elections' topic, while a lot of the Satire articles are grouped in the 'Others' topic. Neither of these two classes contained a lot of articles labelled with the 'Economy' topic.

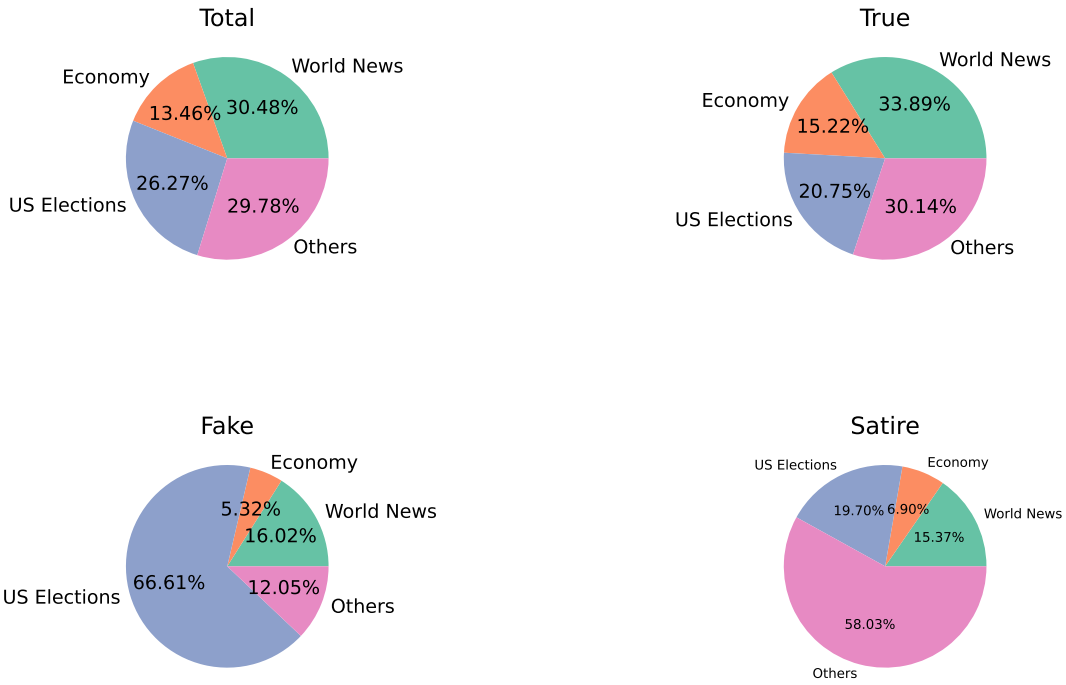



Figure 4.2: Topic Distribution Across Classes

## 4.3 Logistic Regression

The f1-score of the true class drops only by 1% between the trained-on and the unseen subsets, which can be seen on the first column in Table 4.2. For the fake class this drop is around 4%, whereas for satire the performance drops by around 8%. This discrepancy is reflective of the different class sizes of these three classes and shows logistic regression's over-reliance on arbitrary patterns that can be found in the training data.

| **Subset** | **True** | **Fake** | **Satire** |
|---|---|---|---|
| Train | 0.9765 | 0.8785 | 0.8744 |
| Validation | 0.9668 | 0.8381 | 0.7926 |
| Test | 0.9686 | 0.8397 | 0.7959 |

Table 4.2: F1-scores Per Class Per Subset

| **LogReg prediction ↓** | **Fake** | **Satire** | **True** |
|---|---|---|---|
| Fake | 4732 | 73 | 409 |
| Satire | 106 | 2281 | 138 |
| True | 1229 | 865 | 39541 |

Table 4.3: Confusion Matrix of Predictions on Unseen Articles

## 4.4 Interpretability of Logistic Regression

| Topic | True | Fake | Satire |
|---|---|---|---|
| Economy | 0.9827 | 0.7233 | 0.8043 |
| World News | 0.977 | 0.6847 | 0.7574 |
| US Elections | 0.9431 | 0.8906 | 0.7465 |
| Others | 0.9665 | 0.7555 | 0.8169 |

Table 4.4: F1-scores Per Class Per Topic on Unseen Articles

| Topic | Misclassification |
|---|---|
| Economy | 3.31% |
| World News | 4.38% |
| US Elections | 8.10% |
| Others | 6.05% |

Table 4.5: Percentage of Misclassified Articles Per Topic

## 4.5 Random Logistic Regression

Looking at the 10 highest and lowest-weighted words of each class for this randomised logistic regression model, which can be found in Appendix B, only confirms that the model did not learn anything as the words are somewhat uncommon and are not semantically related.

| Random LogReg Prediction ↓ | Fake | Satire | True |
|---|---|---|---|
| **Fake** | 5 | 5 | 61 |
| **Satire** | 0 | 0 | 0 |
| **True** | 29902 | 16223 | 199235 |

Table 4.6: Confusion Matrix of All Predictions of the Random Logistic Regression Model

## 4.6 BERT Results

Both of the fine-tuned BERT models were trained for 5 epochs and both models improved their performance with each epoch. The improvement in performance began to plateau at the fifth epoch, suggesting that further training would only marginally improve performance. The validation loss of the random BERT even drops below the training loss at epoch 4, as seen in 4.3. This is because the model cannot learn any true patterns of the data and almost completely learned to always predict the biggest class. The random nature of the labels results in more of the training predictions being incorrect than predictions on the validation set.

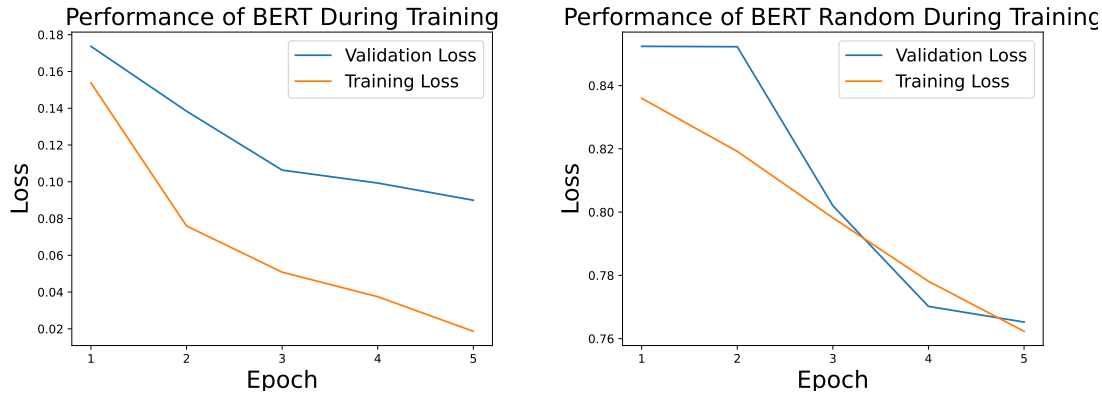



Figure 4.3: Training Curves on Misinformation Detection

| Subset | True | Fake | Satire |
|---|---|---|---|
| Train | 0.9996 | 0.9975 | 0.9993 |
| Validation | 0.9884 | 0.9466 | 0.95349 |
| Test | 0.9888 | 0.9486 | 0.9496 |

Table 4.7: F1-scores Per Class Per Subset of Fine-tuned BERT Predictions

| BERT prediction ↓ | True | Fake | Satire |
|---|---|---|---|
| True | 39749 | 344 | 233 |
| Fake | 279 | 5723 | 10 |
| Satire | 60 | 0 | 2976 |

Table 4.8: Confusion Matrix of BERT Predictions on Unseen Articles.

### 4.6.1 Random BERT

The results of the randomly fine-tuned model are not presented as the model simply learned to classify all articles as 'True', the largest class. Its overall accuracy is 81%, corresponding to the proportion of articles in the True class.

## 4.7 Interpretability of BERT

| Topic | True | Fake | Satire |
|---|---|---|---|
| Economy | 0.9947 | 0.9240 | 0.9626 |
| US Elections | 0.9747 | 0.9558 | 0.9297 |
| World News | 0.9928 | 0.9171 | 0.9518 |
| Others | 0.9905 | 0.9507 | 0.9570 |

Table 4.9: F1-scores of Unseen True, Fake, and Satire Articles Across Different Topics

Despite overall significantly higher performance, we see the same patterns as with the logistic regression model, caused by the imbalance of classes within these topics.

| Topic | Misclassifications |
|---|---|
| Economy | 0.98% |
| World News | 1.32% |
| US Elections | 3.32% |
| Others | 1.58% |

Table 4.10: Percentage of Misclassified Articles Per Topic

## 4.8 Probing Tasks

Below are the distributions of sentiment and verb tense labels across classes, and the descriptions of the sarcasm and clickbait headlines datasets, followed by training and validation loss of different probes based on different BERT models during training, as well as the overall table of results on probing tasks.

### 4.8.1 Sentiment

Only 23,430 articles, just below 10% of the original dataset, were correctly labelled with sarcasm, sensationalism, sentiment and verb tense predictions. For reasons explained in A, the sarcasm and sensationalism labels were rejected from probing tasks and replaced with publicly available datasets of news headlines designed for predicting sarcasm and clickbait. Below is the distribution of positive, negative and neutral labels in different classes. Since most of the True class was labelled as having a positive sentiment, that is the most prominent sentiment overall. The sentiment of Fake articles is especially even and the Satirical articles are less likely to be negative, which could be due to their light-hearted, comedic nature.

| Class ↓ | Negative | Neutral | Positive |
|---|---|---|---|
| True | 2691 | 5891 | 10438 |
| Fake | 1003 | 1000 | 949 |
| Satire | 303 | 549 | 606 |

Table 4.11: Distribution of Sentiment Labels Across Misinformation Classes

### 4.8.2 Verb Tense

Like Conneau et al. [22], I use verb tense as a probing task not intuitively related to misinformation detection. As for the sentiment task, verb tense labels of the same 23,430 articles from the original dataset were used for this probing task. As expected, most news articles are written in the past tense, however, the proportion of present to past tense articles is much higher in the Fake class, at 1:0.425 compared to around 1:0.2 for the True and Satire classes.

| Class ↓ | Past | Present |
|---|---|---|
| **True** | 15924 | 3096 |
| **Fake** | 2071 | 881 |
| **Satire** | 1202 | 256 |

Table 4.12: Distribution of tense labels across misinformation classes

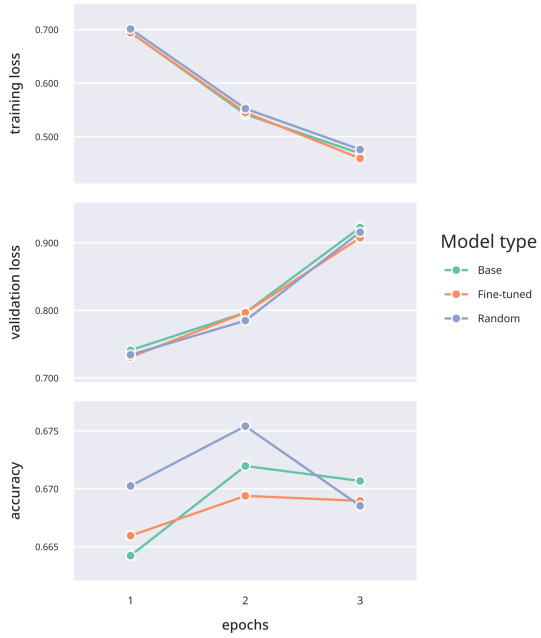



Figure 4.4: Sentiment

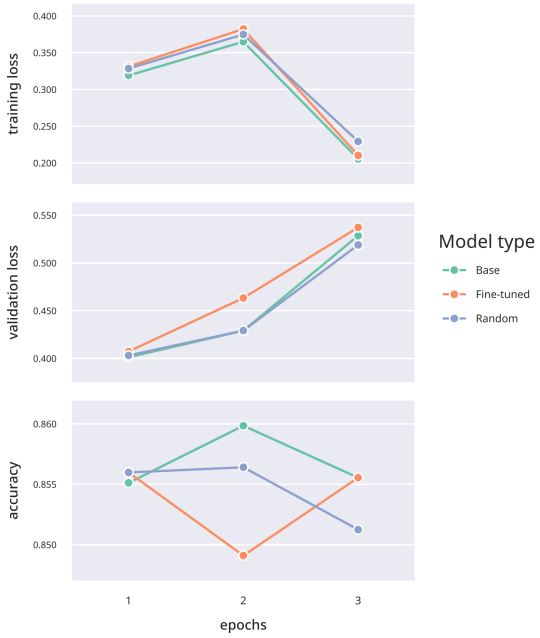



Figure 4.5: Verb Tense

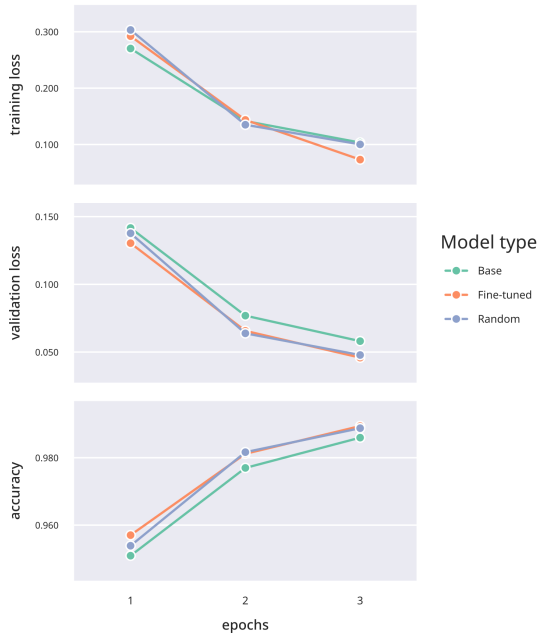



Figure 4.6: Sarcasm

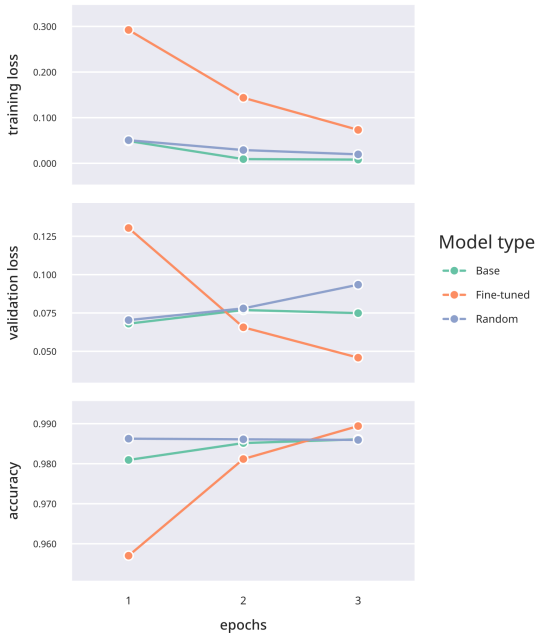



Figure 4.7: Clickbait

| Probing task | Model type | Training loss | Validation loss | Accuracy |
|---|---|---|---|---|
| Sentiment | Fine-tuned | **0.459** | **0.907** | 0.669 |
| | Base | 0.469 | 0.923 | **0.671** |
| | Random | 0.476 | 0.916 | 0.669 |
| Sarcasm | Fine-tuned | **0.073** | **0.046** | **0.990** |
| | Base | 0.103 | 0.058 | 0.986 |
| | Random | 0.100 | 0.048 | 0.989 |
| Clickbait | Fine-tuned | 0.073 | **0.046** | **0.989** |
| | Base | **0.008** | 0.075 | 0.986 |
| | Random | 0.020 | 0.093 | 0.986 |
| Tense | Fine-tuned | 0.210 | 0.537 | **0.856** |
| | Base | **0.206** | 0.529 | **0.856** |
| | Random | 0.229 | **0.519** | 0.851 |

Table 4.13: Performance of Probes After Epoch 3

### 4.8.3 Model Training Curves During Training on Probing Tasks

### 4.8.4 Probe Results are Not Just Correlation

It was ensured that the models' performance on these probing tasks is not merely a reflection of the probe task labels' correlation with the original class labels. Not only is it the case that the sentiment and verb tense labels are approximately evenly distributed across the original classes, as seen in tables 4.11 and 4.12, the models fine-tuned for misinformation detection often predict probe labels correctly for inputs that were misclassified in the original task and vice versa.

On the unseen (validation and test) dataset with sentiment and verb tense labels consisting of 4,594 articles, the fine-tuned BERT predicted the misinformation label incorrectly for 87 articles. 52, or 60%, of those articles' sentiment labels were predicted correctly. In general, 68% of the unseen articles' sentiment labels were predicted correctly. Although the small sample size partially makes this result less reliable, it nevertheless shows that fine-tuned BERT's incorrect misinformation classification is not highly correlated with its performance sentiment probe classification. On the respective training dataset, fine-tuned BERT makes incorrect misinformation predictions for 12 articles, for 10, or 83%, of these, it makes the correct sentiment classification.

Similarly, out of the 87 articles that the fine-tuned BERT model incorrectly predicted the misinformation label, 71, or 82%, were correctly classified by the verb tense probe based on the same BERT model. This is very close to the overall verb tense probe accuracy of 85% on unseen articles. Like for sentiment, 10 out of the 12 articles misclassified by the model in misinformation detection in the training dataset had their tense predicted correctly by the verb tense probe.

These statistics point to the fact that the accuracy on probing tasks is not a result of correlation with predictions on the original misinformation detection task. Although this methods requires more rigorous research to ensure its viability, using different target labels of the original trained-on dataset is a viable way to create probing tasks.

# Chapter 5

# Discussion

This chapter aims to explain why the results presented in the previous section support the hypotheses set out in §1.3. It presents a comparison of predictions made by logistic regression and BERT and analyses patterns in their misclassifications. It then compares the two random models to compare how logistic regression and BERT deal with uninformative training data. It briefly states the ease of interpretability of logistic regression, before extensively analysing results of the three BERT models on the four probing tasks. The chapter concludes by comparing the achieved results to those achieved in related research, presenting the limitations of the approaches and methods used in this project, and giving recommendations for addressing these, and continuing this research in future work.

## 5.1 Distribution of Articles Across LDA Topics

The uneven distribution of topics in different classes, as seen in pie charts in §4.2, can be explained by the fact that the True class dominates the dataset, with 81.20% of the articles belonging to it, as seen in table 4.1. Therefore, when LDA tries to cluster articles into four distinct and approximately evenly distributed topics, it maximises this the most in the largest class. Since the smaller classes differ linguistically, their topic distributions deviate from the True class and hence are skewed. This is confirmed by the fact that the distribution of topics in the True class is the most similar to the overall topic distribution of the whole dataset.

## 5.2 Performance of Logistic Regression and BERT

As hypothesised in §1.3, using BERT for misinformation detection significantly improved classification accuracy, with an overall weighted f1-score of 98.10% and 98.86%, 94.76% and 95.15% f1-scores for the True, Fake and Satire classes respectively, as can be seen in table 4.7. This is an improvement on the performance of the logistic regression model, which was already high, with an overall weighted f1-score of 94.22% and 96.68%, 83.89% and 79.43% on the True, Fake and Satire classes, respectively.

BERT misclassified 926 out of the 49,374 unseen articles. Out of those, 510 are also misclassified by the logistic regression model. In total, the logistic regression model misclassified 2,820 out of the 49,374 unseen articles.

| **Model ↓** | **True - Satire:Fake** | **Fake - Satire:True** | **Satire - Fake:True** |
|---|---|---|---|
| BERT | 0.22 | ∞ | 0.043 |
| LogReg | 0.34 | 0.086 | 0.084 |

Table 5.1: Ratio of Classes Into which Unseen Articles were Misclassified. Based on tables 4.3 and 4.8.

As seen in table 5.1, the logistic regression model is more likely to misclassify articles into the largest remaining class than the BERT model. This highlights the fact that the logistic regression model is relying on simple statistical heuristics to classify articles, while BERT relies on complex linguistic knowledge from its pre-training, enabling the model to correctly classify articles and be less affected by the size of each class.

This shows that deep pre-trained contextual encoders are less prone to be affected by imbalances in class sizes, as shown by the fact that it has a higher f1-score for satirical articles than for fake ones, despite the fake class being almost twice its size. This reflects the fact that satirical articles have a more distinct writing style, whereas fake news attempt to mimic the style of genuine news articles.

This shows the importance of selecting the right model for the task. The use of a more complex model is warranted for this task as the dataset size is large enough to prevent overfitting and the addition of the satire class makes it more linguistically complex. These results support hypotheses 1 and 2, stated in 1.3.

## 5.3 Random Control Models

The performance of the logistic regression model fine-tuned on data with randomly-shuffled labels, presented in 4.6 shows that it approximated predicting the most common class, True, but it did not learn this heuristic perfectly. It attempted to guess the Fake class a few times, reflecting the fact that a sizeable fraction of the training data belonged to this class, hence the model learned some spurious correlations between the articles that were randomly assigned to it. Unsurprisingly, its predictions were random guesses.

BERT, on the other hand, learned to always predict the largest class and hence maximised its accuracy at 81%, but had f1-scores of 0 for the Fake and Satire classes.

Based on this, logistic regression and BERT differ in dealing with uninformative training data. Logistic regression still aligns with whatever spurious correlations are present in the data, while BERT 'understood' that articles with the same labels are not at all linguistically correlated and learned to minimise its loss.

## 5.4 Misclassifications by BERT and Logistic Regression

Despite BERT having significantly higher performance than logistic regression, both models followed the same patterns in misclassifying articles from different topics.

For the simpler mode, the difference in performance between articles labelled with the 'World News' topic (4.38% misclassified), compared to those in the 'US Elections' topic (8.10% misclassified), seen in table 4.5 is striking as the size of these datasets are similar, namely 30.48% of articles are under 'World News' and 26.27% are labelled as 'US Elections'. This discrepancy in performance points to the fact that 66.54% of all Fake articles are grouped in the 'US Elections' topic, compared to just 16.02% in the 'World News' topic and, conversely, 33.89% of all True articles are in the 'World News' topic, compared to just 20.75% in the 'US Elections' topic. Therefore this model's performance on articles of different topics is merely a reflection of its performance on different classes and their distribution within the topics.

The differences between misclassifications made by BERT and logistic regression in different LDA topics, as seen in tables 4.10 and 4.5 are minimal, as both models tend to misclassify articles in certain topics in similar proportions. BERT is also most likely to misclassify articles from the 'US Elections' topic, with a 3.32% misclassification rate (compared to 8.10% for logistic regression (LR)), followed by 'Others' at 1.58% (6.05% for LR), then 'World News' at 1.32% ( 4.38% for LR) and finally 'Economy' at 0.98% (3.31% for LR). Like for the logistic regression model, these differences in performance are due to the distribution of classes between these topics.

## 5.5 Interpretability of Logistic Regression

Logistic regression offers simple explanation for its classification decisions by assigning weights to each word that indicate how much the presence of that word increases or decreases the likelihood of classification into each class. The ten highest and lowest-weight words for all classes can be found in Appendix B.

Unsurprisingly, the highest-weighted words are intuitively associated with their classes, for example "com" and "weblink" are highly associated with the Fake class, while "reuters" and "say" are highly associated with the True class. Interestingly, the highest-weighted words for one class are often some of the lowest-weighted words for other classes. For example, "reuters" is in the lowest-weighted words of the Fake and Satire classes, pointing to how the model learned to classify text solely based on the presence of certain words. Conversely, the words selected by the model trained on data with randomly-shuffled labels are rare, as common words would likely be evenly distributed between these classes, and seem completely random and unrelated to anything.

## 5.6 Probing Tasks

The differences in performance of the three models on the four probing tasks are consistent, but not statistically significant, as can be seen in the results table 4.13 and training curves 4.8.3. Fine-tuned BERT has the lowest validation loss on sentiment,

sarcasm and clickbait prediction, and the highest validation loss on the verb tense prediction task, which is the control probe not intuitively related to misinformation detection. The accuracy of all models on most probing tasks is almost indistinguishable, however, fine-tuned BERT still has the highest accuracy on the sarcasm and clickbait tasks and the same exact accuracy as BERT base on the verb tense task.

The model fine-tuned for misinformation detection has clearly distinct performance 4.13 and training curves 4.8.3 from BERT base and BERT random.

### 5.6.1 Sentiment and Sarcasm Probes

All three models' training curves are virtually indistinguishable when trained on the sentiment and sarcasm probing tasks 4.8.3. However, from 4.13, it's clear that fine-tuned BERT has consistently lower training and validation loss than the other models.

#### 5.6.1.1 Anomalous Training and Validation Loss

It is abnormal that the validation loss is consistently lower than the training loss for the sarcasm task. Since the clickbait headlines dataset is not verified, it is possible that the way it was split up makes the validation set easier than the training set, meaning that the validation set may be more representative of the distribution of the data.

Other possible explanations for this unusual performance, such as data leakage and regularisation methods can be ruled out since it was ensured that none of the headlines in the validation set appear in the training set, and no regularisation methods are used, as for the other probing tasks. This also happens for the fine-tuned BERT model on the clickbait task, but this could be coincidental since the training and validation of all models vary more significantly for this task.

### 5.6.2 Verb Tense Probe

The accuracy of fine-tuned BERT notably drops from epoch 1 to epoch 2 on the verb tense prediction task 4.8.3, which is not the case for the other models. However, at epoch 3 fine-tuned BERT has comparable performance to the remaining models, suggesting that it becomes less confident in its predictions when training for this task as the probe's input embeddings don't contain information as directly related to the verb tense prediction task as the other models. As hypothesised in 1.3, BERT base achieves the lowest training loss and its accuracy increases from epoch 1 to epoch 2. An increase in training loss indicates that the model is making more errors on the training data, but an increase in accuracy suggests that these errors are overall smaller as the models are becoming less confident in their predictions. This could suggest the fact that none of these models create embeddings directly well-suited to this task.

### 5.6.3 Clickbait Probe

The difference in performance between fine-tuned BERT and the remaining models on the clickbait task suggests that this task is more different from the task of misinformation detection than sentiment, sarcasm, and even verb tense prediction. The three models

are most markedly distinguished in this task by their training loss. Despite achieving the best validation loss and accuracy, fine-tuned BERT has the highest training loss of 0.073, while BERT base reaches a staggering training loss of 0.008. This result is anomalous, especially given the relatively high validation loss of 0.075.

Although fine-tuned BERT has significantly higher training and validation loss and lower accuracy than the other two models on this task at epoch 1, it learns faster than these models and achieves a lower validation loss and higher accuracy by epoch 3. This could suggest that the information encoded in the embeddings produced by this model could be more related to this task than those produced by other models, however, the probe had to change its weights more to reach this performance. Whereas the general information encoded in BERT base and BERT random could more easily be used for this prediction task, which is supported by the fact that the training curves for these two models are comparatively linear.

Since the sarcasm and clickbait datasets are completely different from the training data, and the fine-tuned BERT model achieves good performance on both, we know it does not rely on simple properties of the training data for this classification task, but rather generalisable linguistic properties, unlike the logistic regression model.

**These results show that BERT fine-tuned for misinformation detection does encode linguistic features intuitively related to that task** more strongly than other models in its embeddings, as hypothesised in §1.3. This supports the potential use of probes for generate explanations of model classification in terms of measures of which linguistic features they consider.

## 5.7 Comparison to Literature

Comparison to results obtained in existing literature is challenging due to the fact that the dataset was constructed just for this dissertation. However, we can compare to other literature utilising similar methods for misinformation detection, as well as the results of other probing tasks for BERT.

Dipto Das et al. [23] have used linguistic features to classify articles into fake and satire and were able to achieve a highest f1-score of 82.5%. Logistic regression, through self-selecting the most important while fitting, albeit linear, features for the task, was able to achieve a weighted f1-score of 94.22%, showing that manually-selected features, although fully interpretable, are not sufficient for this classification task and more complexity is necessary.

One implementation of BERT for satire detection by Ionescu et al. [37] used CamemBERT (French BERT) and achieved an accuracy of 97.48% on the test set. Kaliyar et al. [40] create a variation of BERT for misinformation detection, FakeBERT, which achieves 98.90% accuracy on the test set. My model achieves a very similar weighted f1-score of 98.10%. However, one would expect a slight drop in performance due to the fact that previous research used binary classes (satire and regular news), while my dataset has three classes. However, these results confirm that deep pre-trained contextual encoders are necessary for highly accurate text classifications.

**Comparison of probe results** is challenging as no previous research has probed a BERT model for misinformation detection, much less the linguistic features of sarcasm or clickbait.

Conneau et al. [22] used verb tense, among many other tasks, to probe different encoding architectures. The best performance on this task, achieved by a GatedConvNet encoder with seq2tree training, achieved 91.5% accuracy. In comparison, the tense probe in this project achieved an f1-score of 85.6%. Chen et al. [19] probe BERT in hyperbolic spaces using sentiment detection. Their BERT base model achieved Spearman's correlation of 91%. Although this result cannot be directly compared to the f1-score of 67.1% as they use binary sentiment, in contrast with the three sentiment classes used in this project.

More research of probes of complex linguistic features is necessary to situate these results and pave the path towards more model interpretability.

## 5.8 Limitations

### 5.8.1 Max_length

One limitation of this dissertation was using `max_length` of 256 for fine-tuning BERT. This means that articles were truncated at 256 tokens, which limited the amount of linguistic information used by BERT as the conclusions of many articles were not included. **Longformer, the long document Transformer** [8], or another architecture, suitable for parsing long variable-length sequences could be used ensure that all the linguistic information in the articles is captured.

### 5.8.2 Probes

There has been a lot of research investigating contextual embeddings, what they encode and how fine tuning a model changes how it encodes text. A comprehensive review of probing tasks was carried out by Belinkov and Glass (2019) [7], where they discuss the promises and limitations of probing for investigating neural embeddings. Particularly, they state that

"*The probing framework may indicate correlations between representations fl(x) and linguistic property z, but it does not tell us whether this property is involved in predictions of f.*"

With this in mind, the results of experiments presented in table 4.13 must be taken as an indication of what information could be correlated to the information encoded in embeddings, not an exhaustive explanation of model decisions.

### 5.8.3 Probing Tasks

Only four hand-picked probing tasks were considered. A more comprehensive review of probing tasks consisting of other linguistic features should be carried out to analyse what kind of information is encoded in BERT. Likewise, only the effect of fine-tuning for the

task of misinformation detection was investigated. Future studies of BERT embeddings should consider analyse the performance of large pre-trained models fine-tuned for various tasks on various related and unrelated semantic probing tasks.

Additionally, using unverified datasets from publicly-available sources makes comparison and evaluation more challenging. As a result, the only evaluation and comparison of these experiments consisted of other models trained for the same tasks. Comparison to existing literature (see §5.7) could not be direct, and anomalous results, such as the§5.6.1.1, may occur.

### 5.8.4 Using GPT-3 for Generating Labels

Employing other large language models for creating ground-truth labels introduces a degree of unreliability to the probing tasks based on them. Wang et al. [78] advocate for this approach to reduce annotation costs and prove the reliability of this approach. Additionally, manual inspection of labels was carried out on tens of randomly-selected articles to ensure that the generated labels aligned with human judgment. However, this still does not ensure that the labels are always correct, especially when the properties being labelled are vague or imprecise, like sensationalism or sarcasm.

### 5.8.5 Random Model Baseline

The training curves of the BERT model trained on data with randomly-shuffled labels closely resemble training curves of the default BERT base model in figures 4.8.3, showing that this model's parameters did not change much as a result of random weight updates. It's possible that they mostly cancelled each other out.

Alternative probing control tasks and models are discussed in the future work section §5.9 below.

## 5.9 Future Work

### 5.9.1 More Comprehensive Probe Metrics

Although probe performance can give an indication of how much the embeddings created for one task correspond with another task, it does not necessarily show that the model in question learns properties for the probing task. A more comprehensive metric of analysis would be Minimum Description Length (MDL), introduced by E. Voita and I. Titov [77]. This metric also conveys how difficult it is to achieve said probe task performance, in regards to the complexity of the probe classifier and amount of training data required, hence it would be a good addition for estimating and quantifying the presence of linguistic features in BERT embeddings. Moreover, the use of simulated control tasks, as explained in a Stanford tutorial on designing and interpreting probes [33], is recommended to verify the validity and robustness of probe performance results.

Another interesting avenue of investigating the information encoded in BERT embeddings is using embeddings at different layers to see whether different types of linguistic

features are encoded hierarchically, as Blevins et al. [10] have found with RNNs.

### 5.9.2 Different Control Models and Tasks

Further research on the effect of fine-tuning BERT in terms of complex linguistic features is needed. Future studies should consider models with randomly initialised weights, as well as models fine-tuned for different tasks. Zhang et al. [80] have studied probe performance on randomly initialised models and found that, given a large enough dataset, its performance is comparable to that of most other model architectures.

Future research should compare probe performance of BERT models fine-tuned for different tasks, or even simulated fake tasks to investigate how much linguistic knowledge is gained or lost through fine tuning in comparison to BERT base.

### 5.9.3 Further Investigations of Complex Linguistic Features

Further research into the contents, and probe performance on other linguistic features, of BERT embeddings fine-tuned for different tasks is required. For example, to investigate whether a conceptual gradient of tasks have performance on linguistic features.

### 5.9.4 Generating Interpretation Based on Probes

Further investigation and proof of the robustness of complex linguistic features in BERT's contextual embeddings would subsequently allow for experiments into creating interpretations for classification decisions.

Using linear probes for various linguistic features, it would be possible to produce measures of the presence of said features in input text (by not applying softmax to the output) found and considered by the model for the classification task.

# Chapter 6

# Conclusion

The three hypotheses presented in §1.3 were supported by results of all experiments.

The task of misinformation detection was successfully expanded to include the class of satire through the creation of a new dataset.

BERT was successfully employed for this task, and achieved a high weighted f1-score of 98.10%, while the logistic regression model achieved an f1-score of 94.22%, confirming the necessity of deep pre-trained contextual encoders for text classification.

There are significant differences in the distribution of topics of articles in different classes. Logistic regression follows linear patterns found in the training data and hence its performance heavily reflects the distribution of articles. In order to ensure that articles of various topics and from various sources are classified correctly, more complex architectures are needed.

Multiple linguistic features with varying degrees of relevance to the task of misinformation detection were used as probing tasks to investigate whether BERT learns to encode, or highlight, linguistic information semantically related to the task it is fine-tuned for through comparison to other models. These experiments weakly confirmed this hypothesis, as the BERT model fine-tuned for misinformation detection achieved consistently, but not statistically significantly, better results than the pre-trained BERT base model, and the model fine-tuned on data with randomly-shuffled labels.

This work paves the path to creating interpretation of classifier decisions through the use of probes corresponding to various linguistic features placed on top of deep textual classifiers.

# Bibliography


[1] Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A Gers. Visbert: Hidden-state visualizations for transformers. In *Companion Proceedings of the Web Conference 2020*, pages 207–211, 2020.

[2] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–236, 2017.

[3] Miguel A Alonso, David Vilares, Carlos Gómez-Rodríguez, and Jesús Vilares. Sentiment analysis for fake news detection. *Electronics*, 10(11):1348, 2021.

[4] Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. Probing linguistic features of sentence-level representations in neural relation extraction. *arXiv preprint arXiv:2004.08134*, 2020.

[5] Oberiri Destiny Apuke and Bahiyah Omar. Fake news and covid-19: modelling the predictors of fake news sharing among social media users. *Telematics and Informatics*, 56:101475, 2021.

[6] Michele Bedard and Chianna Schoenthaler. Satire or fake news: Social media consumers' socio-demographics decide. In *Companion proceedings of the the web*. 2018.

[7] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Comput. Linguist. Assoc. Comput. Linguist.*, 48(1):207–219, April 2022.

[8] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[9] Clément Bisaillon. Fake and real news dataset, March 2020.

[10] Terra Blevins, Omer Levy, and Luke Zettlemoyer. Deep rnns encode soft hierarchical syntax. *arXiv preprint arXiv:1805.04218*, 2018.

[11] Adrian MP Braşoveanu and Răzvan Andonie. Visualizing transformers for nlp: a brief survey. In *2020 24th International Conference Information Visualisation (IV)*, pages 270–279. IEEE, 2020.

[12] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[13] Jason Brownlee. Multinomial logistic regression with python. https://machinelearningmastery.com/multinomial-logistic-regression-with-python/, 2021. Accessed 10/2022.

[14] Cameron Buckner. Deep learning: A philosophical introduction. *Philosophy compass*, 14(10):e12625, 2019.

[15] Clint Burfoot and Timothy Baldwin. Automatic satire detection: Are you having a laugh? In *Proceedings of the ACL-IJCNLP 2009 conference short papers*. 2009.

[16] Judee K Burgoon, J Pete Blair, Tiantian Qin, and Jay F Nunamaker. Detecting deception through linguistic analysis. In *Intelligence and Security Informatics: First NSF/NIJ Symposium, ISI 2003, Tucson, AZ, USA, June 2–3, 2003 Proceedings 1*, pages 91–101. Springer, 2003.

[17] Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. Stop clickbait: Detecting and preventing clickbaits in online news media. In *Advances in Social Networks Analysis and Mining (ASONAM), 2016 IEEE/ACM International Conference on*, pages 9–16, 2016.

[18] Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 397–406, 2021.

[19] Boli Chen, Yao Fu, Guangwei Xu, Pengjun Xie, Chuanqi Tan, Mosha Chen, and Liping Jing. Probing bert in hyperbolic spaces. *arXiv preprint arXiv:2104.03869*, 2021.

[20] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*, 2019.

[21] Jonathan Clarke, Hailiang Chen, Ding Du, and Yu Jeffrey Hu. Fake news, investor attention, and market reaction. *Information Systems Research*, 32(1):35–52, 2020.

[22] Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *arXiv preprint arXiv:1805.01070*, 2018.

[23] Dipto Das and Anthony J Clark. Satire vs fake news: You can tell by the way they say it. In *2019 First International Conference on Transdisciplinary AI (TransAI)*, pages 22–26. IEEE, 2019.

[24] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[25] Michael C Dorf and Sidney G Tarrow. Stings and scams: fake news, the first amendment, and the new activist journalism. *U. Pa. J. Const. L.*, 20:1, 2017.

[26] Tom B. Brown et al. Language models are few-shot learners for prognostic prediction. February 2023.

[27] Facebook. About fact-checking on facebook and instagram. https://www.facebook.com/business/help/2593586717571940?id=673052479947730. Accessed: 2023-4-6.

[28] Elena Filatova. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. *Lrec*, 2012.

[29] Elizaveta Friesem. Hidden biases and fake news: Finding a balance between critical thinking and cynicism. *Social Education*, 82(4):228–231, 2018.

[30] Haomin Gong and Xin Yang. Digitized parody: The politics of egao in contemporary china. *China Information*, 24(1):3–26, 2010.

[31] Daniel Halpern, Sebastián Valenzuela, James Katz, and Juan Pablo Miranda. From belief in conspiracy theories to trust in others: Which factors influence exposure, believing and sharing fake news. In *Social Computing and Social Media. Design, Human Behavior and Analytics: 11th International Conference, SCSM 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26-31, 2019, Proceedings, Part I 21*, pages 217–232. Springer, 2019.

[32] Yaru Hao, Li Dong, Furu Wei, and Ke Xu. Self-attention attribution: Interpreting information interactions inside transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12963–12971, 2021.

[33] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. *arXiv preprint arXiv:1909.03368*, 2019.

[34] Jay D Hmielowski, R Lance Holbert, and Jayeon Lee. Predicting the consumption of political TV satire: Affinity for political humor,the daily show, andthe colbert report. *Commun. Monogr.*, 78(1):96–114, March 2011.

[35] Benjamin Hoover, Hendrik Strobelt, and Sebastian Gehrmann. exbert: A visual analysis tool to explore learned representations in transformers models. *arXiv preprint arXiv:1910.05276*, 2019.

[36] Hugging Face Inc. Trainer - hugging face transformers. https://huggingface.co/docs/transformers/main_classes/trainer#trainer, 2021. Accessed on April 6, 2023.

[37] Radu Tudor Ionescu and Adrian Gabriel Chifu. FreSaDa: A french satire data set for cross-domain satire detection. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, July 2021.

[38] Francisco Jáñez-Martino, Rocío Alaiz-Rodríguez, Víctor González-Castro, Eduardo Fidalgo, and Enrique Alegre. A review of spam email detection: analysis of spammer strategies and the dataset shift problem. *Artificial Intelligence Review*, 56(2):1145–1173, 2023.

[39] Heejung Jwa, Dongsuk Oh, Kinam Park, Jang Mook Kang, and Heuiseok Lim. exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences*, 9(19):4062, 2019.

[40] Rohit Kaliyar. Fakebert: Fake news detection in social media with a bert-based deep learning approach. *Multimedia tools and applications*, 80(8):11765–11788, 2021.

[41] Shashank Kapadia. Evaluate Topic Models: Latent Dirichlet Allocation (LDA). https://towardsdatascience.com/evaluate-topic-model-in-python-latent-dirichlet-allocation-lda-7d57484bb5d 2019. Accessed 12/2022].

[42] Junaed Younus Khan, Md Tawkat Islam Khondaker, Sadia Afroz, Gias Uddin, and Anindya Iqbal. A benchmark study of machine learning models for online fake news detection. *Machine Learning with Applications*, 4:100032, 2021.

[43] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018.

[44] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention module is not only a weight: Analyzing transformers with vector norms. 2020.

[45] Sebastian Kula, Michał Choraś, and Rafał Kozik. Application of the bert-based architecture in fake news detection. In *13th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2020) 12*, pages 239–249. Springer, 2021.

[46] Quoc V Le. Building high-level features using large scale unsupervised learning. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 8595–8598. IEEE, 2013.

[47] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. April 2021.

[48] Or Levi, Pedram Hosseini, Mona Diab, and David A Broniatowski. Identifying nuances in fake news vs. satire: using semantic and linguistic cues. *arXiv preprint arXiv:1910.01160*, 2019.

[49] Lily Li, Or Levi, Pedram Hosseini, and David A Broniatowski. A multi-modal method for satire detection using textual and visual cues. *arXiv preprint arXiv:2010.06671*, 2020.

[50] William Lifferth. Fake news.

[51] Chao Liu, Xinghua Wu, Min Yu, Gang Li, Jianguo Jiang, Weiqing Huang, and Xiang Lu. A two-stage model based on bert for short fake news detection. In *Knowledge Science, Engineering and Management: 12th International Conference, KSEM 2019, Athens, Greece, August 28–30, 2019, Proceedings, Part II 12*, pages 172–183. Springer, 2019.

[52] M. MacCarthy. Government efforts to censor social media should be transparent. Forbes, October 5 2022.

[53] Hadar Marksverdhei. Clickbait title classification dataset. https://huggingface.co/datasets/marksverdhei/clickbait_title_classification, 2021. Accessed: February, 2023.

[54] MathWorks. Choose number of topics for lda model. https://www.mathworks.com/help/textanalytics/ug/choose-number-of-topics-for-LDA-model.html. Accessed 12/2022].

[55] Priyanka Meel and Dinesh Kumar Vishwakarma. Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems with Applications*, 153:112986, 2020.

[56] David Mimno, Matt Hoffman, and David Blei. Sparse stochastic inference for latent dirichlet allocation. June 2012.

[57] Katherine Ognyanova, David Lazer, Ronald E Robertson, and Christo Wilson. Misinformation in action: Fake news exposure is linked to lower trust in media, higher trust in government when your side is in power. *Harvard Kennedy School Misinformation Review*, 2020.

[58] Ellie Pavlick. Semantic structure in deep learning. *Annual Review of Linguistics*, 8:447–471, 2022.

[59] ME Peters, M Neumann, M Iyyer, M Gardner, C Clark, K Lee, and L Zettlemoyer. Deep contextualized word representations. arxiv 2018. *arXiv preprint arXiv:1802.05365*, 12, 2018.

[60] Abdul Qadir. NLP with LDA (Latent Dirichlet Allocation) and Text Clustering to improve classification. https://towardsdatascience.com/nlp-with-lda-latent-dirichlet-allocation-and-text-clustering-to-improve-cl 2020. Accessed 12/2022].

[61] Md Raquiba. Sarcasm news headline dataset. https://huggingface.co/datasets/raquiba/Sarcasm_News_Headline, 2020. Accessed: February, 2023.

[62] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[63] Richard Rogers. Research note: The scale of facebook's problem depends upon how 'fake news' is classified. *The Harvard Kennedy School (HKS) Misinformation Review*, 2020.

[64] Ana-Cristina Rogoz, Mihaela Gaman, and Radu Tudor Ionescu. SaRoCo: Detecting satire in a novel romanian corpus of news articles. May 2021.

[65] Victoria L Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell. Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of*

*the second workshop on computational approaches to deception detection*, pages 7–17, 2016.

[66] Mike Schuster and Kaisuke Nakajima. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, March 2012.

[67] Neha Seth. Part 2: Topic Modeling and Latent Dirichlet Allocation (LDA) using Gensim and Sklearn. https://www.analyticsvidhya.com/blog/2021/06/part-2-topic-modeling-and-latent-dirichlet-allocation-lda-using-gensim-and 2021. Accessed 12/2022].

[68] Reza Shokri, Martin Strobel, and Yair Zick. On the privacy risks of model explanations. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 231–241, 2021.

[69] Blaž Škrlj, Nika Eržen, Shane Sheehan, Saturnino Luz, Marko Robnik-Šikonja, and Senja Pollak. Attviz: Online exploration of self-attention for transparent neural language modeling. *arXiv preprint arXiv:2005.05716*, 2020.

[70] A Smith, L Silver, C Johnson, and J Jiang. Users say they regularly encounter false and misleading content on social media - but also new ideas. *Pew Research Center*, 2019.

[71] Xinying Song and Denny Zhou. A fast wordpiece tokenization system, 2021.

[72] Mateusz Szczepański, Marek Pawlicki, Rafał Kozik, and Michał Choraś. New explainability method for bert-based model in fake news detection. *Scientific reports*, 11(1):23705, 2021.

[73] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. May 2019.

[74] Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316*, 2019.

[75] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[76] Jesse Vig. Visualizing attention in transformer-based language representation models. *arXiv preprint arXiv:1904.02679*, 2019.

[77] Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. March 2020.

[78] Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. Want to reduce labeling cost? GPT-3 can help. August 2021.

[79] Fan Yang, Arjun Mukherjee, and Eduard Dragut. Satirical news detection and analysis using attention mechanism and linguistic features. *arXiv preprint arXiv:1709.01189*, 2017.

[80] Kelly W Zhang and Samuel R Bowman. Language modeling teaches you more syntax than translation does: Lessons learned through auxiliary task analysis. *arXiv preprint arXiv:1809.10040*, 2018.

[81] Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. Fake news: Fundamental theories, detection strategies and challenges. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 836–837, 2019.

[82] Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. June 2015.

# Appendix A

# Rejecting Sarcasm and Sensationalism Labels

Using GPT-3 API to create binary labels for the presence of sarcasm or sensationalism in articles resulted in very low estimations of these classes, rendering fine tuning a classifier with these impossible. Instead,the few articles that were classified as sarcastic or sensationalised are analysed below.

After filtering out 1113 nonsense responses for all new labels (sarcasm, sensationalism, sentiment, verb tense), the only article that GPT-3 labelled as sensationalised is:

"*Cameras catch two drug smugglers crossing the fence from Mexico into Arizona does anyone else out there think we need the National Guard pronto!...*"

The article is Fake and in the 'US Elections' class. It seems that this is a highly sensationalised headline and not a full article. This suggests that GPT-3 didn't label full articles as sensationalised as they contained a variety of sensationalised and down-to-earth language. With lack of access to ground-truth knowledge, it's possible that GPT-3 did not label longer text as sensationalised.

Similarly, only 19 articles were labelled as sarcastic. The articles belonged to a pretty even split of topics; 7 in 'US Elections', 6 in 'Others', and 3 in 'Economy' and 'World News' each. Surprisingly, this small set was made up of 12 True and 7 Fake articles, but no Sarcastic articles. It's difficult to imagine that none of the 1458 satirical articles that were labelled by GPT-3 were sarcastic.

Examples of articles that were labelled as sarcastic:

"*Obama has waisted billions on green energy but the free market wins in the end with the success of fracking and natural gas. You won't hear this from Obama because it doesn't further his agenda to spread the wealth of American taxpayers. Obama plans on spreading our tax dollars to India Yes, India!...*"

"*Do Kansas City sports teams know how to get a postseason party started or what? On the opening kickoff of Saturday's Wild Card Game against the Houston Texans, Chiefs' return man Knile Davis raced 106 yards to the end zone without being touched, let*

*alone tackled ...*"

Although all the articles classified as sarcastic are highly sarcastic, the low classification rate can be explained by poor prompt engineering, poor choice of GPT-3 hyperparameters, both of which were selected after brief experiments, or by the text-davinci-03 model's poor lack of understanding of satire and parsing of text longer than a few sentences.

# Appendix B

# Logistic Regression Interpretability

## B.1 Main Logistic Regression Words

Top 10 highest-weighted (↑) and lowest-weighted (↓) words per class by the logistic regression model:
↑ **Fake**: via, featured, com, weblink, october, wire, by, image, posted, november
↓ **Fake**: say, reuters, solely, picture, reporters, twitterweblink, unlimited, spokesman, monday, commentary
↑ **Satire**: reporters, according, added, reportedly, in, several, explain, spokesperson, announce, resident
↓ **Satire**: accord, via, wednesday, twitter, thursday, reuters, october, image, november, video
↑ **True**: reuters, solely, picture, say, commentary, thursday, opinions, cent, wednesday, accord
↓ **True**: via, featured, according, com, in, shit, added, entire, wire, by

## B.2 Random Logistic Regression Words

Top 10 highest-weighted (↑) and lowest-weighted (↓) words per class by the random logistic regression model:
↑ **Fake**: execute, commentators, uninsured, stitch, romance, aviation, fernandinho, cullen, roles, marc
↓ **Fake**: pirate, potentially, collect, explosives, attacker, juncker, feed, near, link, rubber
↑ **Satire**: ware, sixth, remember, critics, according, dare, burnham, gate, lawson, computer
↓ **Satire**: together, spokeswoman, royal, wogan, israeli, audience, clear, regional, huckabee, clay
↑ **True**: cnn, siblings, permanent, coulter, guantanamo, breastfeed, darfur, mack, corbyn, peddle
↓ **True**: examples, observe, buckeyes, turkmen, roads, terminal, normal, opt, saldanha, cautious

# Appendix C

# Fine-tuning BERT

Two 12-layer `bert-base-uncased` BERT models were fine-tuned, one on the misinformation detection training dataset, another on the same dataset with randomly-shuffled labels. This model is large enough to capture complex linguistic patterns, but doesn't require as much computational power to fine-tune and predict with as `bert-large-uncased`, which has 24 hidden layers. As the misinformation detection task did not involve entity recognition or other reasons why capitalisation should be preserved, the uncased version of the model, which lowercases all input text, was selected. The `max_length` was set to 256 tokens as this is standard practice with BERT, however, this resulted in articles over this token limit being capped at 256. See 5.8.1 for alternatives. The`batch_size` was set to 3 to navigate limited-memory environments. BERT is able to achieve high accuracy with only a few training epochs because its weights are initialized using pre-training on a large dataset, which already encodes complex linguistic features, hence the BERT models for 5 epochs. Devlin et al. [24] use only 4 training epochs , but these models were fine-tuned for 5 to ensure that embeddings were thoroughly changed in order to analyse the differences between their embeddings. The model with the best validation performance is selected for making predictions and further analysis. The standard AdamW optimiser with learning rate 1e-5 and epsilon 1e-8 (to avoid dividing by zero) was used, which is standard for fine-tuning BERT models.

The random BERT model was fine-tuned in the exact same way, except on the training set with randomly-shuffled labels.