An accessible and interpretable approach to fake news detection

Iris Marieke Wiggerts: 494876

ERASMUS UNIVERSITY ROTTERDAM

Erasmus School of Economics

Master Thesis: Data Science and Marketing Analytics

Supervisor: Bas Donkers

Second assessor: Khismatullina, M

Date final version: 30-04-2023

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

Table of Contents

1. Introduction

Although fake news has become a widely discussed topic in the last years, it has existed for a long time, even before the printing press was invented. However, the invention of the press enabled news stories to spread much faster, making it easier for fake news to go "viral". In 1835, the New York Sun published multiple articles about the discovery of life on the moon, which were very widespread. This was later named the "Great Moon Hoax". Much later in 2016 during the presidential elections, fake news was frequently in the headlines of U.S. newspapers. Many fake news stories favoring either Trump or Clinton were spread, and it was questioned whether the spread of fake news influenced the outcome of the elections (Allcott & Gentzkow, 2017). More recently, especially during the COVID-19 pandemic, many fake news stories were circulating on the internet. In such times of uncertainty, it is difficult to assess the veracity of news stories and statements (Apuke & Omar, 2021). The main difference between the distribution of fake news in recent years compared to a long time ago is the rise of social media (Allcott & Gentzkow, 2017). Nowadays, social media enable users to share their ideas easily and interact with each other, while there is often no strict quality control of the ideas that are being shared. At the same time, many national, regional, and local news sites are increasingly allowing applications on their websites that enable user contributions. These user contributions can take the form of texts and images of news events, but also comments, blogs, and community listings (Singer, 2009). In traditional journalism, verifying the validity of the statements that are being put out is an important task. However, when not only journalists but also regular users can produce news content, verifying the validity of the statements becomes a very time-intensive task. Fact-checking sites such as Politifact and Snopes manually evaluate whether statements are true. Yet, manually checking all statements and articles would be too time intensive, which calls for an automated approach to fact-checking. Facebook already uses AI tools to flag content that is likely misinformation (Facebook, 2020). They note that their systems are not working optimally yet, and since claims are often expressed in different ways, they are challenging to categorize.

Algorithms for Natural Language Processing (NLP) tasks that use Transfer Learning from large pre-trained models have become increasingly popular (Sahn et al., 2020). These models often contain hundreds of millions of parameters, making them very complex. Because of the complexity of the models, they require a lot of computational power and memory to run. For large platforms such as Facebook and Twitter, it might be easier to implement complex models to detect fake news, however, smaller news sites or forums might not have the same resources to detect fake news. Sahn et al. (2020) have shown that through knowledge distillation, much smaller models can still achieve a performance that is comparable to that of larger models. Another challenge is how to interpret the output of complex models. From a societal perspective, knowing which parts in an article indicate

that the article is fake might be very useful in a practical setting on news sites. From an academic point of view, it is important to ensure that a model makes predictions based on the right assumptions. It is well known that algorithms can be biased in a way that they discriminate against some population groups (Noble, 2018). To ensure that algorithms are not biased, explanation methods can be used to explain their output. In the context of fake news detection, Shu et al. (2019) and Szczepański et al. (2021) have implemented methods to explain the output of their deep learning models, though the deep learning models that they used to predict the veracity of the news were very large and complex. Since there is a need for models that perform well but also efficiently and are easy to interpret, the main research question of this study is as follows:

How can the detection of fake news become accessible and interpretable?

To answer this research question, DistilBERT will be applied to a real-world fake and true news dataset. DistilBERT is a distilled version of Bidirectional Encoder Representations from Transformers (BERT) and performs well at detecting semantic and long-term dependencies in sentences. Because of the use of self-attention, the model is able to understand that when statements are expressed in different ways, they can mean the same thing. DistilBERT can be used to obtain embeddings of the text. Two types of embeddings, namely the [CLS] embeddings automatically generated DistilBERT and mean pooled embeddings, will be extracted through DistilBERT to test which embeddings work optimally for the data used in this study. These embeddings will then be used to train a Neural Network algorithm and obtain predictions. After, an out-of-sample dataset will be created using ChatGPT (OpenAI, n.d.) to test the trained Neural Network algorithm. Since fake news is often spread by bots (Shu et al., 2017), it is interesting to test whether bot-generated fake news content is easy to detect. The predictions on the ChatGPT data will be explained using Local Interpretable Model-Agnostic Explanations (LIME). DistilBERT and Neural Network are both black box models, meaning that it is hard to understand why DistilBERT gives sentences certain embeddings and why the Neural Network predicts an article to be in a certain class. LIME provides local explanations of the output of black box models, which can give an insight into which words in an article make the article likely to be predicted to be fake.

The remainder of this paper is structured as follows. In the next section, a theoretical framework provides the context in which this research is relevant and discusses state-of-the-art implementations of fake news detection. In section 3, the methods used in this research will be discussed. Section 4 provides a description and analysis of the data. In section 5, the results of the implementation of the models are provided and an analysis of the results is done. Lastly in section 6, the results and their implications are discussed and conclusions are drawn.

2. Theoretical framework

Definitions of fake news

To classify news articles into fake news or true news, it is important to formulate a definition of fake news. Not one universal definition of fake news has been put forward yet, while this could benefit the analysis and detection of fake news. Three properties are important for fake news classification. The first property is authenticity, i.e. if the article is verifiable as false. The second property is intent, i.e. the article is written with the intention to mislead readers. The last property is whether the article is a news article or not. A broad definition, formulated by Zhou and Zafarani (2018) is: "*Fake news is false news*". False news is verifiably false, however, it does not need to be written with the intention to mislead readers. Additionally, this definition captures a broad definition of news, meaning that it can be written by journalists and non-journalists and can take the form of articles, claims, speeches, and posts among other sources of information. Allcot and Gentzkow (2017) provide a narrow definition of fake news as "*news articles that are intentionally and verifiably false, and could mislead readers".* According to this definition, the false information must be written with the intent to mislead readers, it should be verifiable that the information is false and the information must be in the form of a news article. Rubin et al. (2015) define fake news as deceptive news. They identify three types of fake news. The first category is serious fabrications. Serious fabrications are a result of fraudulent reporting, in which claims may be made that have not been verified. Examples of this are articles with ''eye-catching'' titles in tabloids or Yellow journalism. Secondly, fake news can be in the form of large-scale hoaxes. Hoaxes are large-scale fabrications that are deliberately written to seem like true news and can cause harm to victims. Lastly, some fake news is produced with the intent of being humorous. News satire and news game shows are examples of this last category. In this study, the validity of news will be considered according to the narrow definition of fake news by Allcot and Gentzkow (2017).

Besides fake news, there are several other concepts also used in related literature. These concepts can be differentiated according to the same properties, namely authenticity, intent, and whether the information is news. As fake news is often connected to these concepts, categorizing them according to the same three properties can help unify the definition of fake news and thus set a base definition for all research related to fake news. Misinformation is false information that is spread without the intention to deceive and does not necessarily need to be news-related. Conversely, disinformation is spread with the intention to mislead others and is also not necessarily news-related. Fake news can be seen as a news-related type of disinformation. Rumors and conspiracy theories can originate from news events and non-news events and are often hard to verify as true or false. Furthermore, besides fact-based, false information can also be opinion-based (Kumar and Shah, 2018). A large body of

literature is dedicated towards the detection of deceptive opinion spam, which refers to fraudulent reviews that are written with the intent to deceive the reader (Ott et al., 2011).

Fake news spread and creation

Fake news is a multidisciplinary issue, and there are several foundational theories in social sciences, economics, and psychology among others that explain why fake news is created and why it has an impact on our society. Approaching the detection of fake news from a multidisciplinary perspective can help understand which incentives lead to the creation of fake news and what biases result in the spread of fake news. Shu et al. (2017) first discussed several psychological and social foundations of fake news. Zhou and Zafarani (2020) extended these foundations and split the fundamental theories into user-related theories, including theories about social impact, individual impact and benefits, and news-related theories. These theories can support the creation of interpretable and justified models. Consequently, this can enhance the qualitative and quantitative analysis of fake news and improve intervention and prevention techniques. First, incentives for the creation of fake news and prominent actors that create and spread fake news will be discussed. Then, several biases pertaining to the recognition of fake news will be examined. Last, content-related theories, aimed at explaining why fake news content might be different from real news content, will be reviewed.

Fake news creation

Users

Fake news is most often created by human users (Zhang & Ghorbani, 2020). These human users can also be called trolls (Shu et al., 2017). Trolls are users that intend to disrupt communities and provoke other users to get an emotional response. The majority of fake news is created by a small number of people (Meel & Vishwakarma, 2020). These fake news users tend to have short-lived accounts, which helps them to stay off the radar of fake news detection technologies (Allcott & Gentzkow, 2017). This indicates that it might be more useful to detect fake news based on the articles or the way that the news is diffused instead of based on the users who diffuse the news. Further, there are several websites that exist purposely to create fake news and resemble real news sites (Alcott & Gentzkow, 2017). In most cases, the fake news is written by humans and then spread automatically or manually (Zhang and Ghorbani, 2020). However, fake news can also be generated automatically. Yao et al. (2017) fine-tuned a deep neural network model, trained on Yelp review data, that writes fake reviews. They used Amazon Mechanical Turk workers, which are anonymous online workers that complete tasks in return for a small reward, to assess the veracity of the generated reviews.

Motivations

There are two main identified reasons to create fake news (Alcott & Gentzkow, 2017). Since fake news stories are often controversial and receive a lot of attention, people might have a monetary incentive to create fake news. Several of the viral fake news stories about the 2016 US presidential elections were created by teenagers living in a small town in Macedonia, who posted stories in favor of both Donald Trump and Hillary Clinton, and subsequentially earned tens of thousands of dollars (Subramanian, 2017). More recently, during the COVID-19 pandemic, Hassan (2020) suggested that individuals created fake news intending to gain online followers. Another reason to create fake news is for ideological reasons. Some individuals might be racially motivated, as fake news was published to create hate towards Chinese people during COVID-19 (Apuke and Omar, 2021). Further, in the context of the 2016 US elections, creators of fake news might generate positive stories about their preferred candidate and negative stories about other candidates to help the position of their candidate in the election (Allcott and Gentzkow, 2017). Apuke and Omar (2021) also conclude that not enough research on the motivations of individuals behind proliferating fake news has been done yet. Much of the research so far is based on anecdotal reports that only show a partial picture (Allcot and Gentzkow, 2017).

Fake news spread

Users

Although fake news is most often created by humans, it is then often spread by non-human users, such as social bots and cyborgs (Shu et al., 2017). Social bots are software-controlled profiles on social media that post content and interact with other bots as well as legitimate human users (Ferrara, 2016). Not all social bots are created with the intention to spread fake news, such as bots used in customer care. However, these bots can sometimes be harmful if they spread unverified news or rumors. Further, some social bots are created with malicious intentions and aim to spam or mislead other users. Social bots can give the impression that a piece of information is very popular among and endorsed by readers, which could lead other people to think that the information can be trusted. Shao et al. (2018) found that social bots played a disproportional role in the spread of articles from low-credibility sources. Bots target users with many followers through replies and mentions, which can lead to humans resharing these posts. Cyborgs are either human-assisted bots or bot-assisted humans (Chu et al., 2012). For instance, when a human user creates an account on Twitter, they might program a bot to post posts every day, while interacting with other users on the platform themselves. Because cyborgs show both human and automated behavior, they can be very successful in sharing fake news and are hard to detect (Shahid et al., 2022). Besides programming bots to spread fake news, human users might also spread fake news manually.

Motivations

Fake news is not always spread with the intention to mislead others, but people might believe the news to be true and consequentially spread the news. However, to assume that individuals only spread information that they believe to be true would be incorrect. Pennycook et al. (2021) investigated the reasons that users might have spread fake news. Both in the context of political news (Pennycook et al., 2021) and news related to COVID-19 (Pennycook et al., 2020), veracity had little impact on sharing intentions. Sharing intentions of fake news were found to be much higher than the assessment of their truth. Pennycook et al. (2021) tested a *preference-based account*, referring to the partisanship preference, for sharing fake news, and found that around 16% of the political headlines were shared despite being identified as fake news. They further tested the *confusion-based account*, which poses that people genuinely mistake fake news for true news. In support of this notion, around 33% of the fake political headlines were identified as true news and shared. Still, however, these two explanations could not explain the spread of all fake news. Lastly, they tested the *inattention-based account*, which posits that although people have a strong preference for only sharing accurate news, they do not pay much attention to the accuracy of the articles they are sharing as they are distracted by the context of social media. By asking the participants to judge the accuracy of the political headlines before sharing them, the spread of fake news headlines decreased by around 50%. The authors concluded that inattention was a strong factor contributing to the spread of fake news. The argument that users are distracted by the context of social media and may thus spread fake news more easily is supported by the findings of Effron and Raj (2020). They found that encountering a fake headline multiple times reduced how unethical people thought it was to share the headline, even if it was labeled as fake news.

A common belief is that people value partisanship, i.e. news supporting their ideological beliefs, over accuracy (Bavel and Bereira, 2017). Osmundsen et al. (2021) found that individuals who reported hating their political opponents were most likely to spread fake news in order to degrade their political opponents. They tested whether the *ignorance theory*, which is similar to the *inattention-based account*, the *polarization theory*, which is similar to the *preference-based account*, or the *disruption theory*, which poses that users share fake news with the intent to disrupt the existing social and political order, was the main explanator for fake news sharing. They found that the main reason for fake news diffusion was of negative partisanship. Their findings suggest that partisan sharers pay more attention to the political usefulness of news rather than the information quality.

Foundational theories of fake news

If nobody believed fake news, it would not have any impact on our society. However, due to several behavioral biases, “normal vulnerable users” (i.e. users without malicious intent) also unintentionally

engage in the spread of fake news. Fake news is written with the intention to mislead readers into thinking the news is true, so it is not surprising that humans are not very good at differentiating fake news from true news. The psychological biases that facilitate fake news can have an individual impact or social impact (Zhou and Zafarani, 2020).

Humans are subject to several psychological biases that impact an individual making individuals vulnerable to fake news. *Naïve realism* refers to the bias that people believe that they have an accurate perception of reality while others with different opinions are mistaken (Ross and Ward, 1996). *Confirmation bias* leads consumers to trust information that confirms their pre-existing attitudes, beliefs, or hypotheses (Nickerson, 1998). It can be seen as a form of *selective exposure* (Friedman and Sears, 1965), which means that people actively seek out information that confirms their existing beliefs and avoid arguments that contradict their opinions. Further, the *overconfidence effect* (Dunning et al., 1990) leads people to overestimate their judgments compared to objective judgments. These cognitive biases can impede an individual's ability to discern fake news from true news. Interestingly, the presentation of true news to correct for misperceptions insufficiently leads individuals to revise their beliefs (*conservatism bias*, Basu, 1997) and might even increase misperceptions among ideological groups (Nyhan and Reifler, 2010).

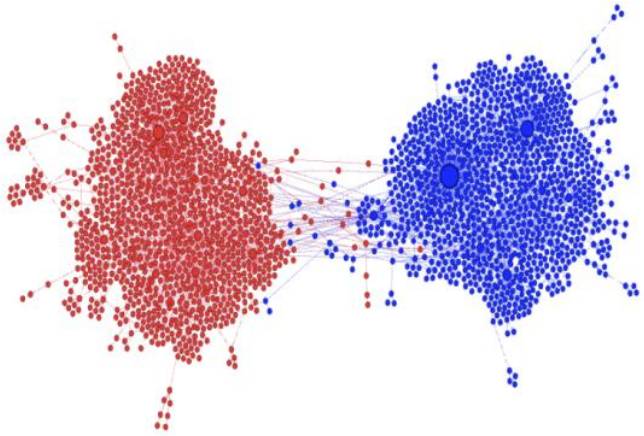


Figure 1: Echo chamber formation in response to the controversial #beefban topic. From: Balancing opposing views to reduce controversy. By: Garimella et al. (2017).

Several psychological biases impact not only individuals but groups of individuals in social settings. The inherent nature of social media can further increase the impact of these biases. For instance, on Facebook, people can form groups with like-minded people in which they are exposed to information that aligns with their already existing worldviews. Additionally, the recommendation algorithms on social media promote information that the user is most likely to read. Both these phenomena can result in an *echo chamber effect* (Jamieson and Capella, 2008). The formation of echo chambers is visualized in Figure 1. Red and blue nodes represent users with opposing beliefs and edges represent retweets on Twitter (Garimella et al., 2017). It can be seen that while there is a lot of interaction

within the groups, not much interaction occurs between the groups, which can lead to tunnel vision, or in other words a narrow point of view. The echo chamber effect exacerbates several psychological biases. Firstly, since news with similar viewpoints is shared many times within these echo chambers, people might be more inclined to believe the information due to the *validity effect* (Boehm, 1994). Further, people derive part of their perception of their identity from being part of certain social groups (*social identity theory*, Ashforth and Mael, 1989). The *bandwagon effect*, denoting that people tend to buy things because other people are buying them, can also be applied to this context, where people post certain articles because other people post them (Leibenstein, 1950). This effect also interacts with the *normative influence theory*, which refers to individuals conforming to the positive expectations of others (Deutsch ad Gerard, 1950).

Content related theories

Several theories justify assuming that, besides discussing fake news as compared to true news, the content of fake news differs from true news. This can be observed in linguistic, semantic and writing style differences between fake and true texts (Zhang & Ghorbani, 2020). Much research on fake news is centered around fake news detection, with the use of fake news and true news content. However, it is important to consider why we should expect to see a difference between fake and true news articles. The following theories will lay the foundation on which news content prediction models, such as the model developed in his study, can be built. It is known that people that are telling lies behave differently from people that are telling the truth (DePaulo et al., 2003). They make a more negative impression and are tenser. Further, they will make fewer ordinary mistakes in their stories and talk less about unusual subjects. Zuckerman et al. (1981) pose that although no one single behavior or cue will always occur when someone lies, the following four factors can be used for deception detection: arousal, control, cognitive processing, and felt emotion. This is also called the *four-factor theory*. The *Undeutsch hypothesis* (Undeutsch, 1967) asserts that statements that are fabricated and not based on real-life experiences differ in terms of content and quality from true statements. The main cause of these differences lies in the fact that experienced events are perceived, and stories about such events will contain more information about the perceptual experience, such as sensory and contextual information. Contrarily, fabricated stories include fewer perceptual details and may contain more information on cognitive operations (Johnson et al., 1993, Johnson and Raye, 1981). It should be noted that the theories stated above relate to deceptive statements or testimonies, and not in particular to fake news. In fake reviews, deceptive experiences are expressed, while fake news does not necessarily relate to an experience. Thus, the theories might be more applicable to deceptive opinion spam as compared to fake news. Zhou and Zafarani (2020) recognize that there is a gap in the literature concerning fake news content theories, but also show

that the same concepts can be applied both to deceptive statements and to fake news. Thus, the same differentiating attributes in the writing style can be extracted to detect fake news.

Fake news detection

Besides survey papers detailing the definition and foundational theories of fake news, most research on the subject of fake news has been directed toward the detection of fake news (Meel et al., 2020). It can be hard for people to recognize fake news as such when they encounter it on social media. To combat fake news, several websites exist that evaluate whether articles or statements are true, such as PolitiFact and Snopes. The people working for these websites determine whether statements are true or false by conducting research. This can be very time-intensive, which is why research towards automatic ways of detecting fake news is valuable. Automatic fake news detection can be done with the use of feature-based or non-feature-based deep learning models. This section will first discuss human efforts to detect fake news and the approaches of several websites and social media sites. Then, automatic fake news detection methods will be examined, first focusing on feature-based methods and then on non-feature-based methods.

Human assessment of fake news

Given the theory on the psychological and social biases that limit individuals in their ability to recognize fake news, it is reasonable to assume that humans will not be very accurate at detecting fake news. Several studies have been conducted to test this hypothesis. Ott et al. (2011) created a dataset including deceptive and truthful reviews with the use of Mechanical Turk workers. Thereafter, three volunteers assessed the veracity of the reviews. The volunteers received an accuracy between 53.1% and 61.9%, indicating that humans do not perform well in identifying deceptive opinion spam. They concluded that the participants suffered from a *truth bias* (Vrij, 2000), meaning that the participants are more likely to classify a review as truthful than fake. They further analyzed that there was little agreement among the volunteers and concluded that the reason for this is that humans are poor judges of deception and focus on unreliable cues for detection (Vrij, 2008).

Kumar et al. (2016) examined the human ability to identify Wikipedia hoaxes, using a dataset consisting of Wikipedia articles that were flagged and then deleted and “normal” Wikipedia articles. They used Amazon Mechanical Turk workers to assess the veracity of the articles. They found that humans can identify Wikipedia hoaxes with an overall accuracy of 66%. From the results, they concluded that humans have a bias toward suspecting that short articles are hoaxes. Further, humans believe that articles with a lower wiki-link density and plain-text-to-markup ratio are more likely to be hoaxes. Thus, the appearance of an article highly influences the judgment of humans, and

when articles are written deliberately to look like a real Wikipedia article it becomes harder to detect its veracity.

Pèrez-Rosas et al. (2017) analyzed how well humans perform in assessing the veracity of celebrity-related news and "general news", containing six domains. They created the general fake news dataset using Mechanical Turk workers for the fake articles and legitimate articles from mainstream news websites. Using two volunteers to judge, they found that humans are better at identifying fake articles in celebrity news than in general news. The accuracy for general news was around 70% and the accuracy for celebrity news was between 77% and 80%. Besides human-written fake news stories, humans also struggle to identify algorithm-created fake news.

Yao et al. (2017) tested how well humans can identify fake restaurant reviews, using a dataset consisting of real reviews and automatically generated reviews. The Mechanical Turk workers in their study achieved a precision of only 40.6% and 16.2% recall. It seemed that workers were more sensitive to repeated errors than to grammar and spelling mistakes. Zellers et al. (2019) developed Grover, a controllable text generator model, that can generate a fake news article if given a headline. Their dataset consists of human-written articles from reputable news sites, Grover-written articles using the same metadata, fake human-written articles, and Grover-written articles using fake metadata. Mechanical Turk workers were used to judge the articles based on stylistic consistency, content sensibility, and overall trustworthiness. They found that humans find Grover-written fake news more trustworthy than human-written fake news, while Grover-written real news is judged as lower quality than human-written real news.

From the aforementioned studies, it can be concluded that humans do not perform very well at judging the veracity of fake news, both written by humans and machines. However, these studies did not investigate how individual differences might affect one's ability to detect fake news. Allcot and Gentzkow (2017) found that people who spend more time-consuming media, people with higher education, and older people have more accurate beliefs about the news. Further, people who use social media as their main source of election news were more likely to believe both fake and real news. Pennycook and Rand (2020) found that susceptibility to fake news is negatively correlated with the tendency to think analytically and conclude that this relationship is due to considerations of headline content. They also found that individuals that overclaim, i.e. individuals that claim they are familiar with something that does not exist, and individuals that tend to rate pseudo-profound sentences as profound are more susceptible to fake news. Bronstein et al. (2019) found that dogmatic individuals and religious fundamentalists are more likely to believe false news.

Automatic detection of fake news

A solution to the low accuracy of human detection of fake news could be to implement automatic detection of fake news. While much research has been published about the detection of fake news using machine learning algorithms, recently non-feature-based deep learning models have become the focus of fake news detection research. The main advantage of deep learning models over feature-based machine learning models is that features are extracted automatically, while feature-based machine learning models require features to be hand-engineered. A more detailed overview of feature-based and non-feature-based detection models used for fake news is provided below.

Fake news detection using feature-based machine learning algorithms

Many machine learning models used for fake news detection rely on feature extraction. The features that are extracted for fake news detection on social media can be related to the news content or related to the social context in which the news articles are posted. Since both feature categories and especially a combination of the two are imperative for fake news detection on social media using feature-based machine learning algorithms, they will be discussed in more detail below.

News content features

News content features describe the meta-information related to a piece of news (Shu et al., 2017). Attributes that are used are the source, headlines, main text, videos, or images. These attributes can be used for linguistic-based feature extraction or visual-based feature extraction. *Linguistic-based features* are extracted from the text in the headline and content of the post, as well as the source. Although fake news is written to mislead people into believing it is real news, often linguistic differences can be observed. As aforementioned, a common motivation for fake news creation is financial gain, so fake news is oftentimes written with the intention of going viral. Therefore, fake news can have “click bait” titles and contain polarizing information and words that are used to exaggerate (Rubin et al., 2015). Hence, even though fake news is written to deceive, linguistic differences might be present in fake and true news, and these features are thus good predictors of fake news. Lexical features are features at the character and word level. These features include total word count, the count of specific words, and the average word length. Syntactic features are extracted that the sentence level. Examples are parts-of-speech (POS) tagging, in which ratios of lexicon markers are reported, and frequencies of function words or phrases, which are often extracted using n-grams and/or term frequency-inverted document frequency (tf-idf) (Rashkin et al., 2017). *Visual-based features* are extracted from visual elements such as videos and images. Photoshop is very accessible nowadays which enables individuals to create fake imagery, while the detection of Photoshop is still challenging (Huh et al., 2018). Further, the rise of deep fakes allows individuals to swap two identities in a single video (Dolhansky et al., 2020). Clearly, there is a need to detect falsities in videos and images. The visual-based features that can be used for deception

detection include image noise and image content features (Zhou et al., 2018) as well as statistical features such as count and image ratio.

Social context features

On social media, social context can be used as additional information for fake news detection (Shu et al., 2017). For instance, in case a malicious user deliberately spreads fake news, recognizing the users that spread fake news might be important. *User-based features* capture the characteristics of users that interact with news articles on social media. These features can be extracted at the individual level or the group level. Individual user-based features can be the number of followers and followed, the age, and the number of posts the user has made for instance (Castillo et al., 2011). At the group level, user-based features are aggregated individual user-based features. It is imaginable, since social media results in the formation of groups and communities, that spreaders of fake news form communities with unique characteristics at the group level (Shu et al., 2017). Besides user-based features, *post-based features* can also improve the detection of fake news. When a news article is posted on social media, users can interact with the post through written replies (Ruchansky et al., 2017) and emojis, such as "liking" a post (Tacchini et al., 2017). Post-based features can be extracted from individual posts, where the topic or credibility can be determined for instance. These features can also be aggregated for all posts related to a news article for insights at the group level. Lastly, *temporal-based features* capture the temporal interactions of users with posts (Ruchansky et al., 2017).

Previous findings

Much research has already been conducted regarding the detection of fake news using feature-based machine learning algorithms. Rashkin et al. (2017) predict fake news based on the Politifact fake news dataset considering a binary outcome class (fake or true), as well as a 6-point outcome class (true, mostly true, half true, etc.). For the binary outcome, a Naïve Bayes model is trained using LIWC measurements concatenated to tf-idf vectors, and an accuracy of 56% is obtained. For the 6-point outcome class, they train a Maximum Entropy model on the same features and obtain an accuracy of 22%. They analyze linguistic differences between satire, propaganda, hoaxes, and true news. They provide several findings. They found that true news uses more assertive words and fewer hedging words than fake news. Also, less reliable news sources use more first- and second-person pronouns compared to more reliable news sources. Further, fake news uses more words that are used to exaggerate, while true news uses more words to present concrete figures. The last two findings are in agreement with the findings of Ott et al. (2011), who apply POS tags to opinion spam. They train a Support Vector Machine (SVM) model on bigrams extracted from opinion spam data and received an accuracy of 89.8%. Most findings are additionally supported by Rayson et al. (2002), who

applied POS tags to imaginative and informative writing in the British National Corpus Sampler. However, they found that comparisons and superlatives were more common in informative writing, which contrasts with the other findings. An explanation for this could be that imaginative writing is different from fake news as it is not written to mislead readers.

Rashkin et al. (2017) additionally found that satire can be differentiated from other types of fake news by its prominent use of adverbs. Rubin et al. (2016) analyzed satire and found that satire often includes absurd sentences, introducing new entities in the last sentences of the satirical news. They train an SVM model on topic-based tf-idf features and grammatical, absurdity, and punctuality features extracted from satire data and achieved an F1 score of 87%. Potthast et al. (2017) applied Unmasking, a meta-learning approach developed by Koppel et al. (2007) using style features such as n-grams and POS extracted from fake news data and satire data. The model predicts fake news with an accuracy of 55% and satire news with an accuracy of 82% (F1 statistic 81%). They found that fake news is more similar to real news than satire is to either one. They showed that it is fairly easy to distinguish satire from fake news. This is contrary to the findings of Horne and Adali (2017), who concluded that fake news is more related to satire than to real news.

Horne and Adali (2017) additionally found that the titles used in fake news are a stronger differentiating factor between fake and real news than the content. Fake news titles are longer than real news titles and contain simpler words in terms of length and technicality. Fake news titles also use fewer stop words, more all-capitalized words, and more proper nouns but fewer nouns overall. Relating to the content, Horne and Adali (2017) found that fake news articles are shorter than real news articles and use fewer technical words, punctuation, and quotes and in general use shorter words. They detect fake news and satire news using an SVM model trained on their top 4 features extracted from the body and title of fake news and satire news. They attain an accuracy of 78% using fake news title data and an accuracy of 90% using satire body data.

Newman et al. (2003) performed Linguistic Inquiry and Word Count (LIWC) on several datasets containing spoken, written, and typed lies and truths. Contrary to previous findings, they found that liars used the first-person pronoun less than truth speakers, which could be because when telling a lie, liars tend to dissociate from the lie (Vrij, 2000). In the context of news, however, more reliable sources might avoid using such pronouns to appear more objective, while fake news sources do not have such considerations. Newman et al. (2003) further found that liars use more negative emotions. Lies also tend to be less cognitively complex, as they contained fewer “exclusive” words and more motion words. Conversely, Castillo et al. (2011) find that tweets with negative sentiment are more likely related to credible news.

Biyani et al. (2016) conclude that the informality on a webpage is a good indicator of how likely it is to be clickbait, where more informality signifies a higher probability. They train a Gradient Boosted Decision Trees model on tf encoded unigram and bigram features extracted from clickbait and non-clickbait news sites that were present on the Yahoo homepage. They receive an F1 score of 75%.

Recasens et al. (2013) analyse biased language on Wikipedia and identify two classes of bias: framing bias and epistemological bias. *Framing bias* occurs when an event is approached from a one-sided stance and often includes subjective intensifiers and one-sided terms. *Epistemological bias* occurs when a proposition commonly thought to be true is questioned, or when an implication is made about a proposition commonly thought to be false. Hedges, entailments, factive verbs, and assertive verbs can signal epistemological bias. Based on their findings Recasens et al. (2013) conclude that the framing bias is more linked to subjectivity in text than the epistemological bias.

Fake news detection using non-feature-based deep learning algorithms

Deep learning is a subfield of machine learning. Although most machine learning models are feature based and thus require manually constructed features, deep learning models extract useful features directly from text input (in the context of fake news detection). The automatic feature extraction is convenient as it makes the model less prone to errors and the predictive accuracy of the model does not depend on the specific features that are used as input. Several studies will be discussed below with particular attention to Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018), as this model will be used in the application of this paper.

Related work

Ma et al. (2016) conducted the first study using a deep-learning model for rumor detection on microblogs. They applied a Recurrent Neural Network model (RNN) to rumor datasets originating from Twitter and Sina Weibo. The RNN model configurated with two Gated Recurrent Unit (GRU) layers performed the best, with an accuracy of 88% on the Twitter dataset and 91% on the Sina Weibo dataset. They deduce that gated units such as GRU can capture long-term dependencies among signals over a basic RNN model. They further notice a higher increase in accuracy in the Twitter dataset when adding the GRU layers to the basic RNN model as compared to the Sina Weibo dataset. The reason for this is that the Twitter dataset contains more noise and extra hidden layers can help overcome noise by capturing high-level interactions more accurately. Ruchansky et al. (2017) develop a Capture, Score, and Integrate CSI model, which captures the temporal pattern of user activity on a post using an RNN, learns the source characteristics, and classifies an article into fake or true. Accordingly, their model captures the text, response, and source properties of fake news. They apply their model to the same datasets as Ma et al. (2016) and obtain an accuracy score of 89% on the Twitter dataset and 95% on the Sina Weibo dataset. Wang et al. (2018) apply an Event

Adversarial Neural Network (EANN) to the same Sina Weibo dataset Ma et al. (2016) and a different Twitter dataset. Their research aimed to develop a model that can identify fake news on newly emerged events. To do this, the EANN model removes event-specific features and keeps shared features among events. Their model received an accuracy score of 72% for the Twitter dataset and 83% for the Sina Weibo dataset. The research of Liu and Wu (2018) has a similar aim, as they propose a model for the early detection of fake news. They do this by classifying news propagation paths with RNN and Convolutional Neural Networks (CNN) and attain an accuracy score of 92% for the Sina Weibo dataset (Ma et al., 2016) and 85% for the Twitter dataset. Nasir et al. (2021) also develop a hybrid CNN RNN model and apply this to two datasets. They find that the combination of two neural network models results in a more generalizable model, meaning it performs better on different datasets overall when compared to other machine learning methods.

BERT

BERT is a bidirectional language model that performs well at detecting semantic and long-term dependencies in sentences. Therefore, it is an appropriate model to use for the detection of fake news with textual data as input. Although BERT is a recently developed model, several applications of BERT for fake news detection exist. Jwa et al. (2019) develop exBAKE, a BERT-based model which the authors pre-train on news data in addition to the regular pre-train data for better representations. ExBAKE further uses Weighted Cross Entropy (WCE) instead of Cross Entropy (CE) to mitigate data imbalance problems. They use a fake news dataset containing headlines and body texts to fine-tune the model and receive an F1 score of 75% with exBAKE compared to 66% F1 with BERT. Kaliyar et al. (2021) propose FakeBERT, a BERT-based model that combines BERT with three parallel blocks of 1D-CNN with different kernel-sized convolutional layers with different filters. The construction of FakeBERT is effective for large-scale structured or unstructured text and can handle ambiguity in the text. The authors apply FakeBERT to a real-world fake news dataset by Ahmed et al. (2018) and attain 98.9% accuracy. Szczepański et al. (2021) emphasize that although deep learning methods such as BERT perform very well, due to their complex nature it is important to understand why they classify an article as fake or true. This way, we can prevent biased models that are based on wrong assumptions. The authors combined BERT with Local Interpretable Model Agnostic Explanations (LIME) and Anchors, which are two Explainable Artificial Intelligence (xAI) techniques. They applied their model to the fake news dataset by Ahmed et al. (2018) and received an F1 score of 98%. Using LIME and Anchors, they determined that the model focused on designations in titles, such as “Turkey” and “Syria”. When these geographical entities were mentioned, the model was very likely to predict the title as true. Conversely, the model also placed a great importance on names, such as “Obama”. When names were used in the titles, they were more likely to be predicted as fake. They

conclude that surrogate models such as LIME and Anchors can capture meaningful patterns and that multiple surrogates should be used since each can have differing insights. Rai et al. (2022) used BERT-base to attain [CLS] embeddings and fed these into a Long Short-Term Memory (LSTM) layer. This construction benefits from the high performance of BERT while the LSTM layer helps memorize and find relevant information patterns. The BERT-based model was applied to the headlines of the FakeNewsNet dataset (Shu et al., 2020), which contains data from PolitiFact and GossipCop. They achieved an accuracy of 89% compared to 86% with BERT on the PolitiFact data and 84% compared to 83% with BERT on the GossipCop data.

3. Methodology

In this study, a similar architecture to the one developed by Rai et al. (2022) will be used. The data used for prediction is first cleaned. Then, the text is tokenized using a DistilBERT tokenizer. After, the pre-trained DistilBERT model (Sahn et al., 2019) is used to obtain embedding outputs, which are then, together with the Class variable from the original dataset, used as inputs for the classification model. For classification, an Artificial Neural Network model (ANN) of which the hyperparameters are optimized using random grid search is used. To interpret the output of the ANN model, LIME is implemented. Raw sentences are used as input for the LIME explainer, which are then fed through a probability prediction function consisting of all the other steps. Using the input sentence and the probability prediction function, LIME provides a local explanation of the model's prediction. A schematic overview of the proposed methodology is shown in Figure 2 below.

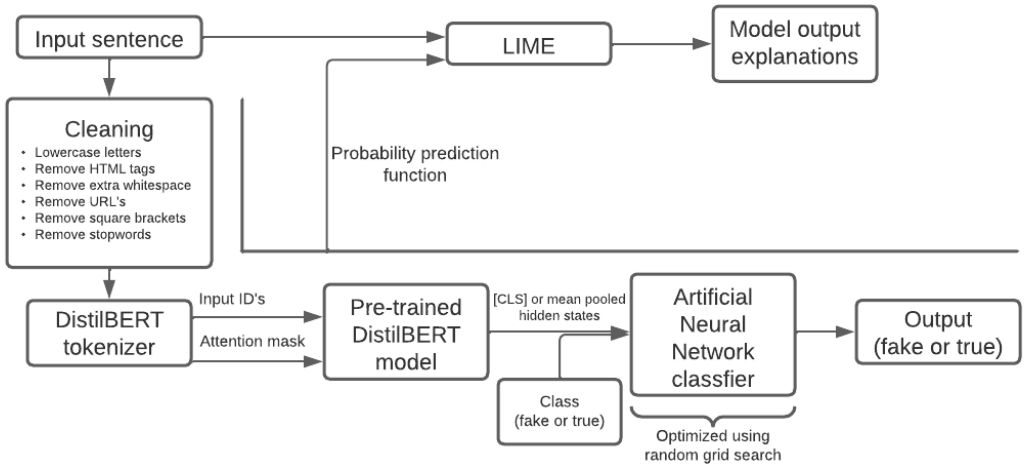


Figure 2: Architecture of proposed methodology.

In this section, first an introduction to transfer learning will be provided. Then, a more detailed description of all components of BERT will be provided, after which the version of BERT used in this study, DistilBERT, will be introduced. Next, the prediction model, the Artificial Neural Network will be

introduced and the hyperparameter tuning technique that was used will be discussed. Lastly, a description of LIME and its implementation on textual data will be given.

Transfer learning

Deep learning models may use sequential inductive transfer learning (Pan and Yand, 2010). The main aim of transfer learning is to boost the performance of the target domain by utilizing source domain data, eliminating the need to collect training data every time a model is trained. This is accomplished by pre-training the model first, in which a general representation of inputs is learned that can be used for different domains. These language representations can be learned by using unidirectional or bidirectional language models. Many current language models analyse text sequences in a unidirectional manner, i.e. from left to right or from right to left. The Generative Pre-trained Transformer by Radford et al. (2018) is an example of such a unidirectional model, as it employs a left-to-right Transformer to predict a text sequence word-by-word. According to Devlin et al. (2019), such unidirectional language models are suboptimal for sentence-level tasks, especially when applying a fine-tuning based approach to a token-level task. In such a task, it is important to incorporate context from both directions. ELMo (Peters et al., 2018) concatenates two independently trained right-to-left and left-to-right unidirectional language models. Still, combining two unidirectional language models does not result in a deep bidirectional model. BERT overcomes the unidirectionality limitation by using a bidirectional training approach. This gives BERT the ability to detect long-distance dependencies in sentences and capture semantics.

After pre-training, the general representations are adapted to a specific task. Two strategies exist for the adaptation: fine-tuning and feature extraction. In feature extraction, which ELMo uses, task-specific architectures are used which include pre-trained representations, and the model's weights are frozen. In the fine-tuning approach, which is used by the Generative Pre-trained Transformer by Radford et al. (2018), minimal task-specific architectures are used and all the pre-trained model's parameters are fine-tuned to a new task. Peters et al. (2019) show that fine-tuning performs well for closely aligned tasks such as semantic textual similarity and next sentence prediction, while feature extraction performs better for distant tasks such as language modelling or sentence pair tasks.

BERT

Bidirectional Encoder Representations from Transformers is a bidirectional language model. It alleviates the unidirectionality constraint of many language models by using a masked language model (MLM) pre-training objective. This objective enables the representation to fuse the left and the right context, resulting in a deep bidirectional model.

Input and output

Figure 3 below shows a schematic overview of the pre-training and fine-tuning processes of BERT. Bert aims to present a single sentence or a pair of sentences in one token sequence, to be able to do tasks such as question-answer prediction or response selection. Thus, the sequence can refer to a single sentence or a pair of two sentences. The input token sequence representation of BERT which is shown in Figure 3, is additionally shown in more detail below in Figure 4. The input embeddings are the sum of the *token embedding*, meaning which word or classification token, *the segment embedding*, denoting in which sentence the token occurs, and the *position embedding*, showing the position of the token in the token sequence. Each token sequence starts with the special classification token [CLS], which denotes the whole sequence representation. In addition to the sentence embedding, two sentences in a single sequence are also differentiated by the [SEP] token, which occurs in between two sentences in a single sequence. The output of the pre-training for token [CLS] is the final hidden vector $C \in \mathbb{R}^H$, and the output of the pre-training for the i^{TH} input token is the final hidden vector $T_i \in \mathbb{R}^H$. Using the output of the pre-trained BERT model, different embeddings such as the mean or max pooled embeddings can also be calculated.

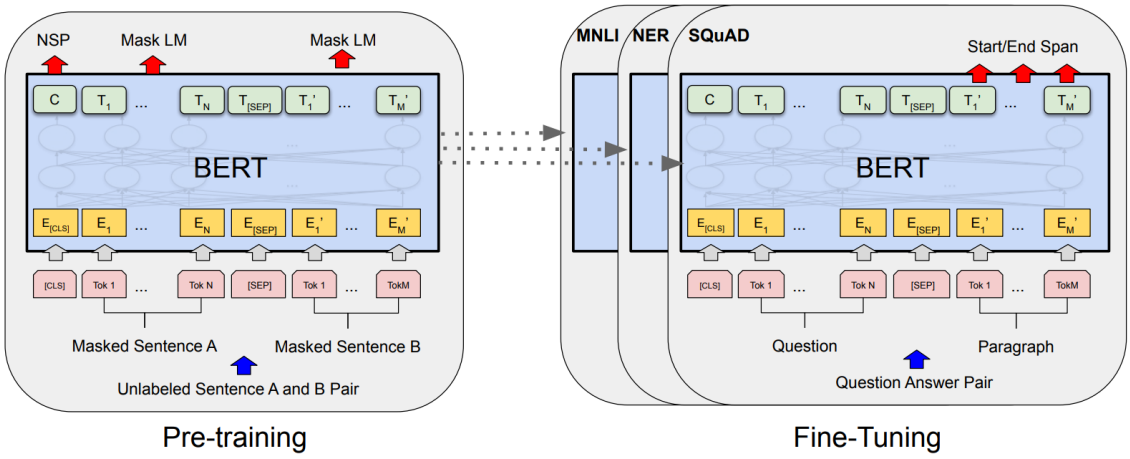


Figure 3: Schematic overview of BERT's pre-training and fine-tuning processes. From: Bert: Pre-training of deep bidirectional Transformers for language understanding. By: Devlin et al. (2018).

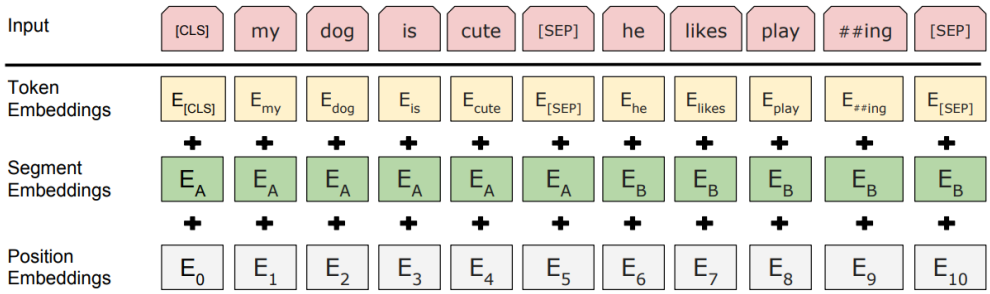


Figure 4: Embedding input BERT. From: Bert: Pre-training of deep bidirectional Transformers for language understanding. By: Devlin et al. (2018).

Transformer

As can be viewed in Figure 3, the architecture for the pre-training and fine-tuning stages are identical, except for the output layer. BERT uses a Transformer with bidirectional self-attention for both pre-training and fine-tuning that is based on the Transformer described by Vaswani et al., 2017. The BERT Transformer consists of six encoders. The encoder maps an input of symbol representations into a sequence of continuous representations. Each encoder is broken into a multi-head self-attention layer and a feed-forward neural network layer. First, the input words are turned into a vector of size 768 using an embedding algorithm. These embeddings then flow through both layers in the encoder and the output of this is the input for the next encoder. The self-attention layer allows the Transformer to take other positions in the input sequence into account for better encoding of a word. For instance, in the sentence "The airplane needed to be repaired because it was broken", the self-attention layer can associate "airplane" with "it". The attention function maps a query and a set of key-value pairs to an output, which are all vectors. The output equals a weighted sum of the values, in which the weight is dependent on a compatibility function between the query and key vectors. Multi-head attention linearly projects the query, key, and value vectors multiple times with different projections and dimensions. It then performs the attention function on each of these projections in parallel and concatenates the multidimensional output values.

Pre-training

The pre-training of BERT uses the BookCorpus (800M words) and the English Wikipedia (2,500M words). An advantage of pre-training is that not many parameters need to be learned from scratch. The data is document-level since this allows BERT to detect long-term dependencies, which would not be possible at the sentence level. The pre-training is done using two unsupervised tasks which will be explained in detail below.

Masked Language Model (MLM)

Contrary to ELMo, BERT does not solve the unidirectionality limitation by concatenating two unidirectional language models. BERT's bidirectional model is more powerful than any unidirectional language model or the concatenation of two unidirectional language models. The reason that most language models are unidirectional, is that in case a model would be bidirectional, words would be able to indirectly "see themselves". This would result in the model being able to predict words in the training sample but only because they are "seen" already. BERT overcomes this problem by randomly masking about 15% of the input tokens (individual words) and then predicts those words. This process is called the masked language model. As shown in Figure 3, the T_i tokens are used for MLM. If token T_i is chosen for prediction, 80% of the time this token is replaced by the [MASK] token, 10% of the time by a random token, and 10% of the time by the unchanged token T_i. This is to limit the

disparity between the pre-training and fine-tuning since the [MASK] token is not present in the fine-tuning. After the token has been replaced, it will be used to predict the original token.

Next Sentence Prediction (NSP)

In addition to language modeling, BERT aims to be able to perform downstream tasks such as question answering or Natural Language Inference (NLI). To do this, relationships between sentences should be analyzed. BERT does this through the Next Sentence Prediction task. The NSP works as follows: for two sentences A and B, in 50% of the cases sentence B follows sentence A and receives the label IsNext, and in 50% of the cases B is a random sentence that does not follow A and receives the label NotNext. BERT's approach differs from representation learning objectives since all parameters are transferred to initialize end-task parameters.

Fine-tuning

Fine-tuning is done according to the specific downstream task that BERT needs to perform with the use of self-attention in the Transformer. The inputs and outputs can be swapped for the different downstream tasks. For instance, Sentence Pair Classification Tasks require two sentences as inputs and output class labels, while Question Answering tasks require a question and a paragraph containing the answer as input and output the start/end span of the answer. The fine-tuning step is computationally inexpensive compared to the pre-training step. This is another advantage of BERT since the pre-trained outputs are open source, limiting the total computation time of an application of BERT.

DistilBERT

Since BERT was released, different versions of BERT have been developed such as RoBERTa (Liu et al., 2019) and DistilBERT (Sahn et al., 2019). In this study, DistilBERT will be used, as it generates faster results and requires a smaller computational power than BERT-base. Rather than using all token embeddings for a fine-tuning task, DistilBERT only uses the embeddings of the [CLS] token during fine-tuning, which represents the aggregate sequence representation for classification tasks. DistilBERT is a distilled version of BERT. Distillation is the transferring of knowledge from a large model, the teacher, to a smaller model, the student, which is faster than the larger model and has a comparable performance (Hinton et al., 2015; Buciluǎ et al., 2006). Neural networks often use a SoftMax layer at the end of the architecture to generate class probabilities. The formula of the SoftMax layer is provided below. The logit, z_i, of each class is converted into a class probability, q_i, by comparing it to the sum of all the exponential scores for all the possible outcomes.

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)}$$

Where T is the temperature that controls the softness of the probability distributions or the smoothness of the output distribution. During training, a high T is set for both the teacher and the student, while after training the student uses a T of 1 to recover a standard SoftMax. The student is trained using the following formula:

$$_{ce} = \sum_i t_i * log(s_i) ,$$

Where the difference between the teacher's predicted probabilities t_i and the student's predicted probabilities s_i is calculated, and the student is penalized for making incorrect predictions. The distillation loss L_{ce} is then combined with the masked language modelling loss L_{mlm} (Devlin et al., 2018) and the cosine embedding loss L_{cos} for the final training objective.

DistilBERT contains 66 million parameters while BERT-base contains 110 million parameters. Although DistilBERT has the same general architecture as BERT, the token-type embeddings and the pooler are removed, and the number layers are reduced by a factor of 2. The dimension size has a smaller impact on efficiency, which is why the dimension of 768, as in BERT-base, is used. Because DistilBERT contains less parameters, it requires fewer data than BERT to be pre-trained. Additionally, DistilBERT is 60% faster than BERT-base and retains 97% of the performance of BERT-base on sentiment analysis of the GLUE benchmark dataset.

Neural network

After obtaining the token embeddings through DistilBERT during pre-training, rather than fine-tuning the embeddings, they were used as input for an artificial neural network (ANN). The embeddings were used as the independent variables, while the class (fake or real) was taken from the original dataset and used as the dependent variable. Using these data as input, predictions on a test set were done by the ANN. Artificial neural networks were originally designed to mimic the neural networks of the human brain (Dongare et al., 2012). ANNs consist of an input layer, multiple hidden layers, and an output layer. The input layer nodes pass the information that is being fed to the model to the first hidden layer. Hidden layers perform computations on the output of the previous layer.

The computations are done using a function z with input x and weight w plus bias b, which is calculated by summing all the products of inputs and weights plus the bias. The bias is a constant that helps the model to fit optimally. The result can be represented as a vector dot product, where n is the number of inputs for the node. The output that is provided in the output layer is the probability for each class of the dependent variable, in the case of a categorical dependent variable. This process is called forward propagation, meaning that given input and weights, an output is computed.

To train the weights, backpropagation is performed. This is done using a function where $* W_X$ is the new weight, W_X is the old weight, a is the learning rate. The learning rate can be seen as the step size to take towards global minima and thus controls the speed of the backpropagation. A small learning rate could result in a very long convergence time, while a large learning rate could result in no global minima being reached. The process of calculating new weights and errors from the new weights and then updating the weights continues until global minima is reached and loss is minimized.

The activation function that is part of each neuron is a formula that determines whether the neuron should be activated or not, depending on whether the input of the neuron is relevant for the prediction. The activation function introduces nonlinearity to the data, enabling the neural network to detect nonlinear patterns, such as high-order polynomials, in the data. Activation functions are monotonic, differentiable, and quickly converge for optimization of the weights. They can be linear or non-linear. Examples of non-linear activation functions are Sigmoid, Tanh, and Maxout.

Random grid search

Artificial neural networks have many hyperparameters that can be tuned to optimize the training of the model to the dataset. Different hyperparameters are important for different datasets, and the hyperparameters might interact in non-linear ways. To find the optimal hyperparameters, a random grid search was conducted (Bergstra & Bengio, 2012). The search for hyperparameters is conducted over a search space, where each dimension represents a hyperparameter and each point represents one model configuration. A grid search is very reliable in low-dimensional spaces, as a grid provides good coverage in a 2-dimensional space, while it has insufficient coverage in subspaces. A random grid draws independently from a uniform density from the same space as a regular grid. The points that are drawn by a random grid search are less evenly distributed in a 2-dimensional space, while they are more evenly distributed in subspaces. Therefore, the random grid search is more reliable in high-dimensional search spaces. Further, the calculations for a random grid search are less exhaustive than a grid search and are less time intensive.

Several hyperparameters were tuned in the random grid search applied in this study. The activation function, the size and the number of hidden layers, and the adaptive learning rate were tuned. Further, the input layer dropout ratio (Srivastava et al., (2014) L1 (Park & Hastie, 2007) and L2 (Cortez et al., 2012) were tuned. These are common regularisation techniques that lower the complexity of the model during training and prevent the model from overfitting. This improves the generalization of the model. L2 encourages weight values towards zero, while L1 encourages the weight values to be exactly zero. These smaller weights reduce the weights of hidden neurons. The hidden neurons become neglectable, which reduces the overall complexity of the neural network. Less complex

neural networks avoid modelling noise in the data, and thus avoid overfitting. Dropout means that with a certain probability, a neuron is turned off during training. This also results in a simpler model.

LIME

Local Interpretable Model-Agnostic Explanations (LIME) (Ribeiro et al., 2016) enables the interpretation of individual predictions made by a black box model. LIME is model agnostic, meaning that it can provide uniform explanations for any black box algorithm. It allows for local interpretation of a model, by explaining how the features relate to individual predictions. To produce an explanation, the surrogate model LIME approximates a black box model, such as a random forest or neural network, locally with a model that is easier to interpret, such as a decision tree or a linear regression (Mishra et. al., 2017). This easily interpretable model is locally faithful to the black box model. However, a pitfall of LIME is that if the underlying model is highly non-linear, even in the local prediction there may not be a faithful explanation. To produce an explanation that is interpretable and locally faithful, the following formula is optimized:

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

Where G is a class of potentially interpretable models, $\Omega(g)$ is a measure of the complexity of $g \in G$, f is the model to be explained, $f(x)$ is the probability that x belongs to a certain class, $\pi_x(z)$ is a proximity measure between an instance z to x, which denotes the locality around x. $\mathcal{L}(f, g, \pi_x)$ is a locally weighed loss function that measures how unfaithful g is in approximating f in the locality π_x, and is trained using perturbated data. As the samples are weighted by π_x in the function above, LIME is fairly robust to sampling noise. The formulation can be used with different explanation families G, fidelity functions $\mathcal{L}$ and regularisation terms Ω.

$\mathcal{L}(f, g, \pi_x)$ is trained to be locally faithful to f by drawing random nonzero sample instances around x, weighed by π_x. The locally weighted square loss function $\mathcal{L}(f, g, \pi_x)$ is optimized by solving the following formula:

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in \mathrm{Z}} \pi_x(z)(f(z) - g(z'))^2$$

Where

$$g(z') = w_g \cdot z'$$

Given the perturbed sample z', the sample in the original representation z is recovered and $f(z)$ is obtained. This is then used as the label for the explainer model. For each word, a probability label and the feature weight is calculated. Given the dataset Z of the perturbed data, the formula is optimized to generate the explanation $\xi(x)$. When LIME is used to explain models that use textual data as input, the perturbed samples are created as follows. The dataset is represented with binary

features for each word. New data is created by randomly removing words from the original text. When a word is included in the local model, it receives a one and otherwise it receives a zero. This is done multiple times, uniformly and at random, to create perturbations. For each perturbed sample the probability and weight is calculated. The weight is the proximity of the sample to the original sample, and is calculated as 1 minus the proportion of the words that were removed.

To generate an explanation of the predictions on the text data, raw input sentences were used, as well as a probability prediction function. This function consists of all steps that the data goes through to generate a probability, which encompasses the following: the data is first cleaned, then tokenized, fed through the DistilBERT model, whereafter the needed output is sliced, which is then, together with the Class variable from the original dataset, used as input for the trained ANN, which then provides a prediction. After feeding the raw input sentence through the probability function, LIME explains the prediction by looking at relative importance of the words in the input sentence and their direction. The explanation is a bag of words, where a limit K is set on the number of words such that

$$\Omega(g) = \infty \mathbb{1}[\| w_g \|_0 > K]$$

In this study, K is set to 10, meaning that the 10 words with the highest explanation weights and their directions are shown to explain the prediction.

4. Data description

The data used in this study originate from Ahmed et al. (2018). The authors collected real-world data for both the fake and true articles. The true news articles were collected from Reuters.com, which is the website of a large news agency Reuters. The fake news articles were collected from websites that Politifact, a fact-checking website, identified as unreliable. The original dataset consists of 23,502 fake and 21,417 true news articles. The news articles are all related to politics and were posted between 31 March 2015 and 19 February 2018. The dataset contains the title, the text, the subject, the day that the article was posted, and whether the article is false or true for each article.

Data preparation

Before the data was used in the DistilBert model, it was cleaned to get rid of invaluable information. Several steps were taken to clean the data in the text and title columns. First, all words were lowercase. Then, HTML tags, extra whitespace, square brackets, and URLs were removed. Lastly, stop words such as *in, a, for, the,* and *is* were removed. After following these steps and taking a closer look at the data, it was noticeable that almost all news articles started with *(Reuters)* and many fake news articles contained *(video)*. In case these words would be included in the data, it could result in a biased algorithm that predicts fake articles well when the fake article contains *(video)*, but also predicts true articles to be fake when they contain *(video),* and the same applies to *(Reuters)* in true

articles. Therefore, these terms were removed from the dataset. Further, several fake and true news articles had duplicated titles and texts in the dataset and were given a different subject. These duplicated articles were deleted from the dataset since they could also result in a biased algorithm. Some fake and true news articles had different titles but the same text, as the text column was empty. These articles were also deleted from the dataset. The resulting dataset contained 17,394 fake and 21,189 true news articles, and a total amount of 38,583 articles. The dataset was thus slightly imbalanced, but this was accounted for in the prediction model.

Data analysis

After the data preparations were done, a closer look was taken at the dataset. Figure 5 below shows the subjects of the news articles differentiated by class. It can be seen that all fake news articles fall either into the category of *politics news* or *world news.* The true news articles have the subjects of *news, politics, U.S. news, left-news,* or *government news*. As the articles in both classes do not have subjects that overlap, this could suggest that the articles are inherently different from each other, which could result in a biased model. It would be problematic if the algorithm can only predict fake articles to be fake in case they have the subject of *politics news* or *world news.* Hence, we dive deeper into the content of the fake and true news articles to see if they differ much from each other.

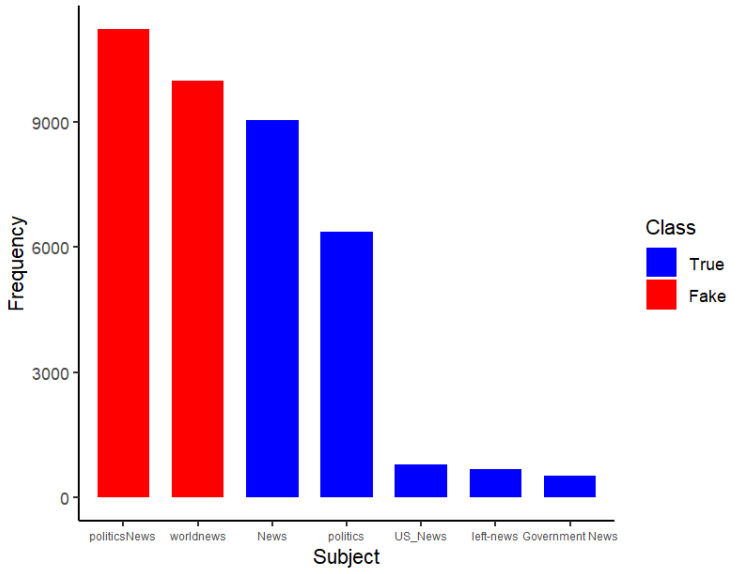


Figure 5: Article subjects of fake and true news articles

Below in Figure 6 and Figure 7, the top ten most common words in true and fake news articles are shown respectively. The words *trump, said, would,* and *president*, occur in both graphs. It is noticeable that the names of politicians, such as *donald, hillary,* and *clinton* occur more often in fake news texts, while in true texts words that refer to institutions such as *government, state, states,* and *house* occur more often in true news texts.

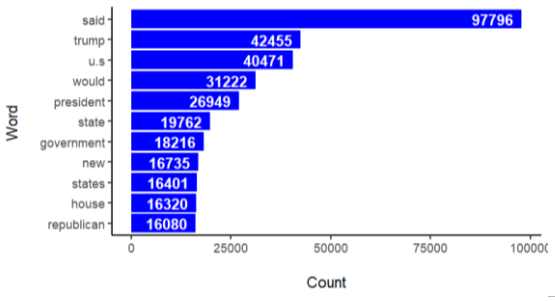


Figure 6: Most frequent words in true news text

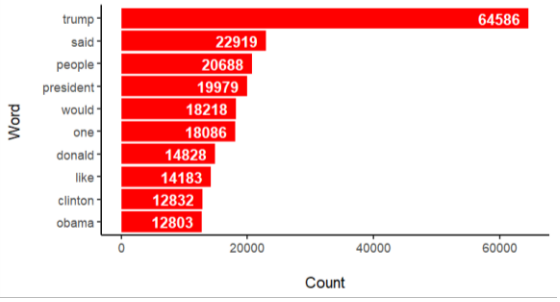


Figure 7: Most frequent words in fake news texts

As shown in Figure 6 and Figure 7 above, the word counts of the most common words in true news texts seem higher than the word counts of the most common words in fake news texts. Further, there is some overlap of words in both text classes. Figure 8 below shows the frequency of the top 10 most used words in fake news texts compared to all words in the texts for fake and true news texts. This figure shows a good comparison of how often the most common words in fake news texts are used both in fake news texts and in true news texts. Even though most words appear relatively more in the fake news texts, the words *said, president,* and *would* which are in the top 10 of both fake and true news texts, appear relatively more in true news texts.

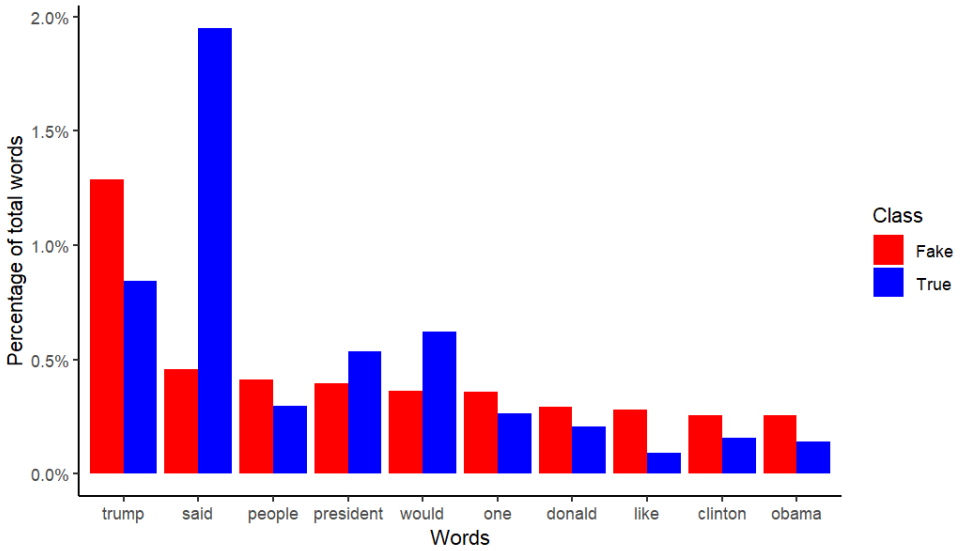


Figure 8: Relative wordcounts for the top 10 most used words in fake texts

Figure 9 and Figure 10 show the distribution of the number of characters in true and fake news texts respectively. It can be seen that many true news texts do not contain very many characters, while longer true texts appear at a higher frequency than fake news texts.

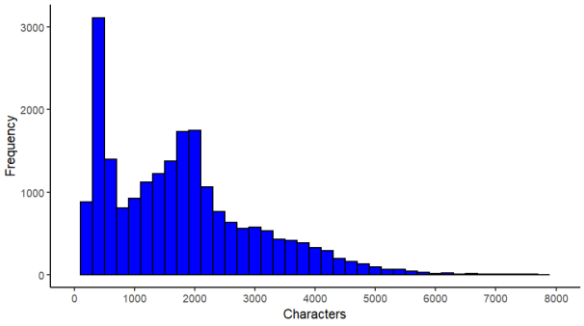


Figure 9: Characters in true texts

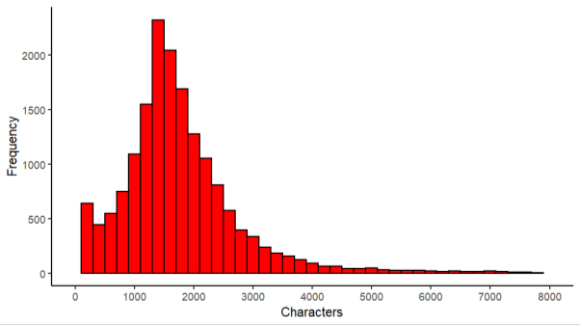


Figure 10: Characters in fake texts

Figure 11 and Figure 12 below show the distribution of the word counts per text for true and fake news texts respectively. It can be seen that even though both fake and true news texts most often contain 200 words, the frequency of this amount of words is higher for fake news texts. Moreover, shorter texts are relatively more common in true news texts than in fake news texts.

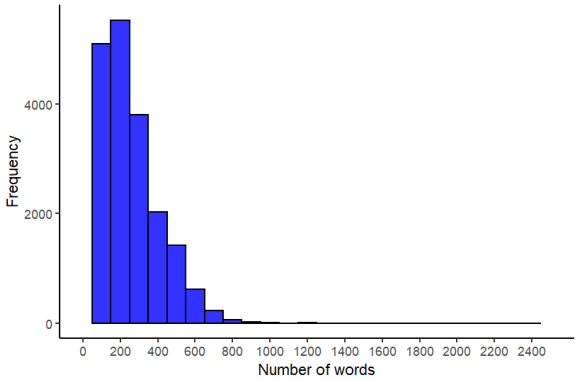


Figure 11: Distribution of wordcounts in true texts

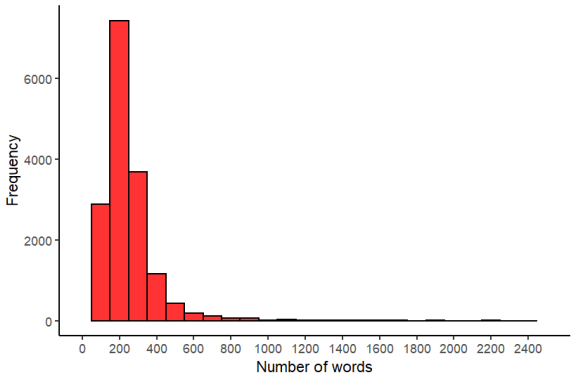


Figure 12: Distribution of wordcounts in fake texts

Figure 13 and Figure 14 below depict the distribution of the average word length of the words in true and fake news texts respectively. Although the distributions seem similar, it seems that the average word length in true texts is slightly larger, while the distribution of the average word length in fake texts is broader.

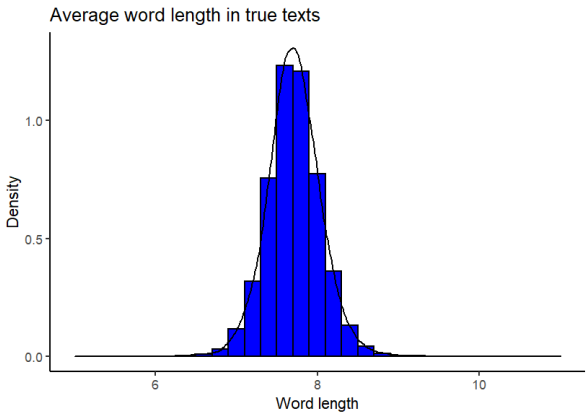


Figure 13: Distribution of average word length in true texts

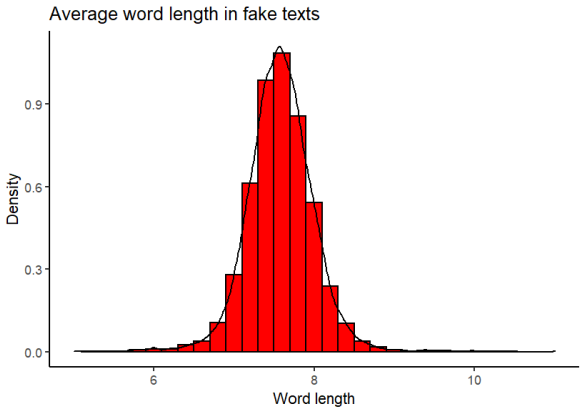


Figure 14: Distribution of average word length in fake texts

5. Results and analysis

Models with several inputs were run to assess which model was optimal to detect fake news articles. According to Rubin et al. (2015), fake news titles often contain polarizing information, and Horne and Adali (2017) found that titles were a better differentiator between fake and true articles. Further, since the maximum sequence length of the input of DistilBERT was set to 50, it was hypothesized that the title input might contain more information than the text input when the inputs are constrained. This is why both the text and the title of the articles were used in separate models for classification, and their results were compared. Additionally, for the text and title outputs, besides the [CLS] token embeddings, the mean pooled token embeddings of all output tokens in a sentence were also extracted. While the [CLS] token embeddings represent the aggregate sequence representation for classification tasks, the mean pooled token embeddings represent the mean sequence representation for classification tasks. The mean pooled token embeddings are calculated the same way as sentence vectors in the Word2Vec mechanism (Mikolov et al., 2013), namely by averaging the word vectors for the words contained in the sentence. In the case of DistilBERT this means taking the element-wise arithmetic mean of the token-level embeddings. In the resulting models, it will be examined whether the [CLS] token which is normally used for classification in DistillBERT performs better as a feature than the mean pooled token, which is based on the benchmark mechanism Word2Vec. The four resulting models will be the model trained on [CLS] text embeddings, the model trained on [CLS] title embeddings, the model trained on mean pooled text embeddings, and lastly the model trained on mean pooled title embeddings.

A more detailed description of the model trained on [CLS] text embeddings is provided in this section. After the dataset was cleaned, the data was tokenized. First, the words were broken into tokens, then, [CLS] and [SEP] tokens were added, and last, the tokens were substituted by their IDs, so that each sentence was represented by a list of tokens. The maximum sequence length of the tokens was set to 50, as the model would otherwise require too high computational power. The tokenized text was padded, meaning that in case a tokenized text contained less than 50 tokens, zeros were added until the total sequence length was 50. Then, an attention mask that withholds performing attention on padding token indices was created and both the attention mask and the padded tokens were fed into the DistilBERT model. As only the [CLS] tokens were used for classification, these were sliced from the output of the model. For the models trained on the mean pooled embeddings, the mean pooled embeddings were calculated and sliced from the output of the model. This resulted in a dataset of 38,583 rows, belonging to the total amount of articles, and 768 columns, belonging to the hidden size of the embeddings. To this dataset, the Class variable from the original dataset was added, which was used as the dependent variable in the neural network model.

The dataset was then separated into a 60% training set and a 40% test set. Then, the training set was separated into a 75% training set and a 25% validation set. These were used in a random grid search of 50 models to find optimal parameters for the neural network model that was used for predictions. The parameters that were tuned were the *activation layer*, which was set to Tanh, the *hidden layers size*, which was set to (20,15), the *adaptive learning rate*, which was set to TRUE, the *input layer dropout ratio*, which was set to 0, and the *L1* & *L2 regularization*, which were both set to 0. Further, *balance classes* was set to true since the data was slightly imbalanced. The resulting deep model consisted of an input layer, two hidden layers, and an output layer.

Below in Table 1, Table 2, Table 3, and Table 4, the confusion matrixes of the predictions of the four resulting models are shown.

Table 1: Text [CLS] token embeddings

	True	**Fake**
True	8272	93
Fake	203	6864

Table 2: Text mean pooled embeddings

	True	**Fake**
True	8143	121
Fake	332	6836

Table 3: Title [CLS] token embeddings

	True	**Fake**
True	8095	452
Fake	380	6505

Table 4: Title mean pooled embeddings

	True	**Fake**
True	8062	376
Fake	413	6581

Table 5: Accuracy scores for all models

Model	Accuracy
Text [CLS]	**98,1%**
Text mean	97,1%
Title [CLS]	94,6%
Title mean	94,9%

As can be seen in Table 5 above, the model using the [CLS] text embeddings performs the best at detecting fake and true news articles. This model predicts the most actual fake articles to be fake and predicts the least actual fake articles to be true. It seems that the mean pooled embeddings, compared to the [CLS] embeddings, perform less well on text input and better on title input. Further, the title embeddings perform less well than the text embeddings overall. The model using [CLS] text embeddings will be used for further analysis.

ChatGPT application

To test whether the deep learning model predicts the articles well because of the inherent differences between fake and true news articles or because of differences between fake and true news articles in our dataset that are not present in other fake and true news articles, predictions on

new data were done. This data was generated by ChatGPT (OpenAI, n.d.). The data consists of 14 articles, of which some are fake and some are true. The topics of some articles are the same as the topics in our large data set, namely *Politics News (US), US Government News, World News,* and *Left News*. ChatGPT was also asked to generate some fake and true news articles with random topics and provided articles with the following topics: *Science/Environment, Health/Technology, World News/Environment, Business/Entertainment,* and *Business/Politics*. The texts written by ChatGPT are notably shorter than the texts in the original dataset. Further, in several fake news texts written by ChatGPT, a false statement is given first and later debunked. It can thus be concluded that the fake and true news articles are not very similar to the articles in the original dataset. However, it is still interesting to see whether the algorithm might still detect fake and true news on the same topic that is written differently and whether the algorithm might detect fake and true news on different topics than the training data. Therefore, they will be used for classification and a deeper analysis at the article level. The same data preparation was used for this dataset, which was then used for the DistilBERT text [CLS] token embedding extraction. After, the dataset was used as a test dataset for the model with the text [CLS] token embeddings. The confusion matrix of the performance of the model with the text [CLS] token embeddings on the new data can be seen in Table 6 below.

Table 6: ChatGPT test data prediction results

	True	**Fake**
True	4	1
Fake	4	5

Table 7: ChatGPT test data prediction results by text subject

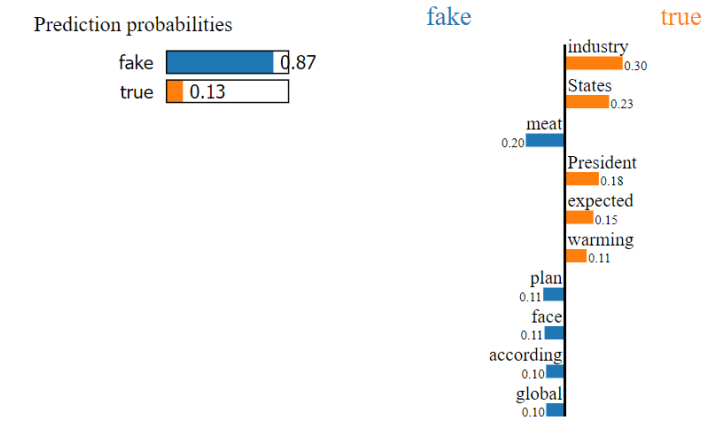
	Class	**Correctly classified**
Politics News	Fake	Yes
Politics News	True	No
US Government News	True	No
US Government News	True	No
World News	True	Yes
World News	True	Yes
Left News	Fake	No
Left News	Fake	Yes
Left News	Fake	No
Science/Environment	True	No
Health/Technology	Fake	Yes
World News/Environment	True	Yes
Business/Entertainment	Fake	Yes
Business/Politics	True	Yes

The accuracy on the test set by ChatGPT is much lower than the test set taken from the original dataset, namely 64,3%. Most fake articles were correctly classified as fake, while half of the true articles were incorrectly classified as fake. In Table 7 it can be seen that the model performs relatively well on the articles with subjects that differ from the dataset on which the model had been trained, which is surprising. While in the training dataset, all articles with the subject World News

were fake, the true World News articles in the test set were correctly identified as true. Additionally, while in the training dataset, all US Government News articles were true, the true US Government News articles in the test set were incorrectly classified as false. Although this test set is too small to draw final conclusions, and it could be that the articles written by ChatGPT on the same subject still differ from the training set, it does not seem to be the case that all Politics News and World News articles are automatically classified as fake and all other subjects are automatically classified as true.

LIME

To further understand why the deep learning model classifies some articles as fake and others as true, LIME will be used to explain individual predictions. Below in Figure 15 the LIME output for an article on the subject of Politics News that was predicted to be fake and is actually fake can be seen. On the top left, the prediction probabilities for both classes are shown. On the top right, the ten words with the highest prediction weights are shown, and towards which class they influence the prediction. On the bottom, the whole article text is shown and the words of the top right illustration are highlighted according to their weight and class. The algorithm is very certain that the article belongs to the class fake, as it was given a prediction probability of 87%. Remarkably, the words with the highest weights for the prediction are words that increase the probability that the article is true. Further, the two words with the highest weights, “industry” and “States”, both relate to institutions, which was also seen in the most common words for true articles in the data description. For the words that indicate that the article is fake, several things can be observed. The word “meat” could be seen as a controversial word, as many people have strong opinions about eating meat, and as can be seen in Figure 1 there are many conversations about it online. Further, “plan” and “face” are both verbs and could be seen as a call to action. Lastly, “according” is often used when talking about an opinion rather than facts.



Text with highlighted words

President Biden is planning to ban meat in the United States, according to anonymous sources within the White House. The sources claim that Biden sees meat as a major contributor to climate change and wants to reduce its consumption to combat global warming. The plan is expected to face significant opposition from the meat industry and some Republicans in Congress.

Figure 15: LIME output for a correctly predicted fake article on the subject of Politics News

In Figure 16 below, the LIME output for an article on the subject of Left News that is predicted to be true and is actually fake can be seen. It is remarkable that even though a wrong prediction was given, the prediction probability for true is 100%. Almost all words that have a high weight for the prediction steer the prediction towards true. It is interesting that even though names of politicians were used very often in fake news articles in the original dataset, as shown in Figure 7, in this article "Sanders" is an indicator that the article should be predicted as true. Further, both in the article in Figure 15 and in this article, "States" is an indicator that the article should be predicted as true.

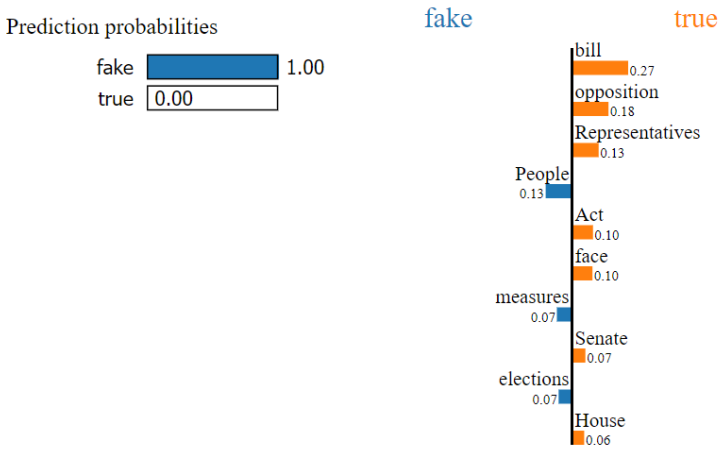
Text with highlighted words

Senator Bernie Sanders has introduced a bill in Congress that would abolish private property in the United States, according to a leaked draft of the legislation. The bill would require all property to be owned and controlled by the government, and would ban individuals from owning homes, cars, or other personal property. Critics have denounced the bill as a socialist takeover of the US economy.

Figure 16: LIME output for an incorrectly predicted fake article on the subject of Left News

Figure 17 below shows the LIME output for an article on the subject of Politics News that was predicted to be fake but is actually true. It is remarkable that even though the prediction is incorrect,

the prediction probability of the fake class is 100%. Even though the article is predicted to be fake, many words that weigh the most in the prediction are words steering the prediction toward true. “Senate” and “House” are both words related to institutions, and “bill” seems to be an important indicator for a true text both in this article and the article in Figure 16.

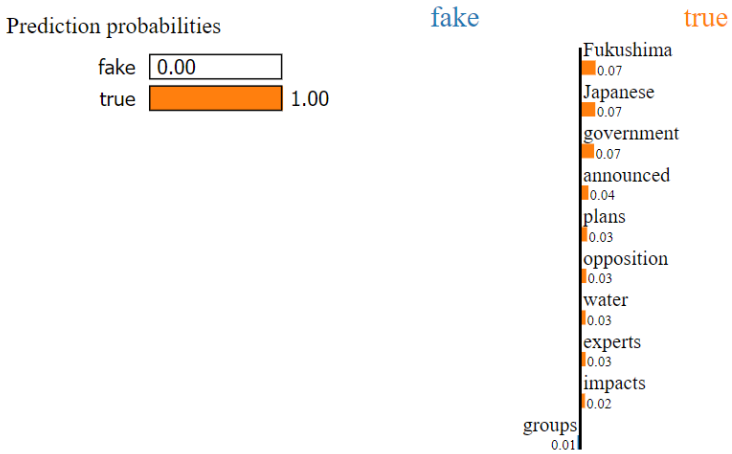


Text with highlighted words

The US House of Representatives has passed a bill to expand voting rights, which would make it easier for Americans to vote in federal elections. The bill, called the For the People Act, includes provisions for automatic voter registration, early voting, and mail-in voting, among other measures. The bill faces an uncertain future in the Senate, where it is expected to face opposition from Republicans.

Figure 17: LIME output for an incorrectly predicted true article on the subject of Politics News

Figure 18 below shows the LIME output for an article on the subject of World News/Environment that is predicted to be true and is actually true. Similar to the other articles, the prediction probability of the predicted class is 100%. Also similar to the other article in Figure 15 that was predicted to be true, almost all words with a high weight for the predictions are words that steer the prediction towards being true. “government” relates to an institution, “Fukushima” and “Japanese” are both geography related words. Further, “opposition” is an indicator of the text to be true both in this article and in the article in Figure 17.

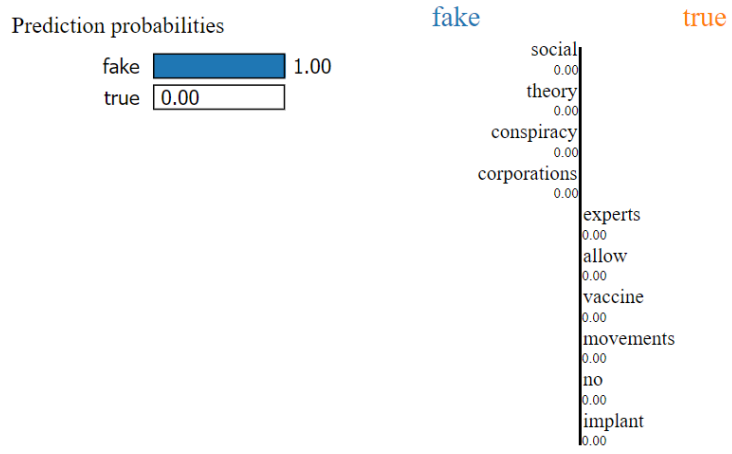


Text with highlighted words

The Japanese government has announced plans to release treated water from the Fukushima nuclear power plant into the Pacific Ocean, despite opposition from local fishermen and environmental groups. The water has been treated to remove most radioactive elements, but some environmental experts warn that the release could still have negative impacts on marine life and public health.

Figure 18: LIME output for a correctly predicted true article on the subject of World News/Environment

Figure 19 below shows the LIME output for an article on the subject of Health/Technology that is predicted to be fake and is actually fake. Similar to the other articles, the prediction probability of the predicted class is 100%. However, none of the words that have the highest weights for the prediction have a high weight, as they all show 0.00. The words "conspiracy" and "theory" both steer the prediction toward fake, which is understandable since fake news articles are often conspiracy theories (Rubin et al., 2015). Further, both in this article and in the article in Figure 18, "experts" is seen as an indicator of a true article. This could indicate that in true articles, references to experts on the topic are made more often.



Text with highlighted words

COVID-19 vaccines contain microchips that allow governments and corporations to track people's movements and activities, according to a conspiracy theory that has been circulating on social media. However, there is no evidence to support this claim, and experts say that it is impossible to implant a microchip in a vaccine. The vaccines have been rigorously tested and shown to be safe and effective in preventing COVID-19.

Figure 19: LIME output for a correctly predicted fake article on the subject of Health/Technology

From the LIME output of several articles, it can be concluded that the same words in different articles often relate to the same class, meaning that the algorithm is consistent in how it labels the words. Even though the output from LIME is instance-based, making it hard to draw conclusions about the overall model, still some things could be observed. Words indicating that the article is true often relate to institutions, geography, and experts. Words that indicate that the article is fake are more controversial and relate to conspiracy theories. Further, in almost all articles considered the prediction probability of the predicted class is 100%, even though some predictions are incorrect. Especially for the articles with different subjects than the training data, it would be imaginable that the prediction probabilities are more distributed across the categories since the texts are different from the texts in the training set, however, this is not the case. This could indicate that the algorithm is overconfident in its predictions. However, it is not very surprising that the LIME prediction probabilities for the predicted class are 100% for most articles, since this was also found in the predictions done by the artificial neural network. For the articles that have different subjects than the training data, it is questionable whether they are classified based on the right inputs. By training a deep learning model on texts with these subjects and comparing the LIME output of such a model to this model, this could be checked. It should also be noted that as the models used to create embeddings and predictions are highly complex, even the local regions might be non-linear, making it hard to receive accurate explanations through LIME.

6. Discussion and conclusion

This paper aimed to design a methodology for accessible and interpretable fake news detection. This was done by combining a DistilBERT and Neural Network model, and assessing the output with LIME. The DistilBERT model implemented in this study contains less parameters than most other pre-trained language models such as BERT-base and RoBERTa. Yet, the model received an accuracy score of 98%, which similar to the score Szczepański et al. (2021), who implemented a complexer model on the same dataset. For a smaller model such as DistilBERT, less data-points are needed to pre-train the embeddings, the computational power required to implement the model is smaller, and the computing time of the model is shorter. These three factors make the model more accessible to smaller organizations, such as regional newspapers or news forums.

Besides developing a model with a focus on accessibility, the methodology was also designed to be explainable and interpretable. This was done by implementing LIME to explain individual predictions. LIME gives an insight into why an article was predicted in a certain class, and which features were influential for this decision. Comparing the influential features for correctly classified articles to those for incorrectly classified articles can show whether the model selected features that might not have been very helpful for incorrect classifications. The outputs of LIME did not show a large difference in

the features that were used in incorrectly and correctly classified articles, as the same features were important in multiple predictions and they only indicated the article to either be fake or be true across articles. In the analysis of the most used words in fake and true texts, words denoting institutions seemed more apparent in true articles and names of politicians seemed more frequent in fake texts. The findings in this study are similar to the findings of Szczepański et al. (2021). They used LIME and Anchors to explain predictions made by a trained Distilbert model using fake news title input from the same dataset as used in this study. They found that the model concentrated on designations, and titles containing these are likely to be true. They also found that names in titles were influential for predictions, and that titles containing these were likely to be false. The LIME output in this study showed that words denoting institutions were important features for some predictions and aligned with the classification true, which could be expected. In two articles of which the output was shown, names of U.S. politicians were present, however in one article the name was not an important feature and in the article the name was steering the prediction towards being true, which is unexpected. This could potentially be explained by the fact that the articles written by ChatGPT differed from the articles in the training set. The articles by ChatGPT were shorter and in some fake articles, a fake claim was debunked at the end of the test. In addition to articles on the same subjects as the training set, ChatGPT also wrote articles on different subjects. Relatively more articles on different subjects than articles with the same subjects as the training set were predicted correctly. Since the test sample was very small, an application including more articles on different subjects should be done to derive final conclusions about how well the model generalizes across contexts.

The findings of this research contradict the findings of the research of Horne and Adali (2017), who found that the titles of news articles were a better differentiator between fake and true news. However, they handcrafted complexity, psychology and stylistic features to conduct their analysis, which could explain the difference in the findings. Both models with title inputs and models with text inputs were tested in this study. The highest scoring text input model received an accuracy score of 98,1%, while the highest scoring title input model received an accuracy score of only 94,9%. Interestingly, in text input models, the [CLS] token embeddings worked better, while in title input models, the mean token embeddings performed better. Although there is no clear consensus on which pooling methods to use, testing multiple pooling methods on the same dataset and selecting the best one can improve the accuracy of the predictions. An interesting direction for future research could be to combine the [CLS] or mean pooled embeddings from the text and title input. Then, it could be tested whether this improves the performance of the model, and which embeddings work the best for this model.

The main limitation of this research was the lack of a processing unit that could handle computations that require a large computational power. To decrease the amount of data to be processed by DistilBERT, the maximum sequence length was set to 50. This means that the model only took into account the first 50 tokens, while most texts were tokenized into more tokens than 50. A better processing unit could enable a higher sequence length for the input of DistilBERT. The model restricts the sequence length to 512, which is more than 10 times larger than the input used in this study. Although the model used in this study attained a high accuracy score, using longer token sequences as input could improve DistilBERTs ability to detect semantics and long distance dependencies in texts. This could then result in a model that is more generalizable. What could also enhance the generalizability of the model is to include news articles on broader topics in the training dataset. However, because fake news is by nature hard to detect, not many datasets on fake news are currently available.

Another limitation was that the stop words that were removed from the text during the data cleaning. This might have impeded DistilBERT from optimally learning semantics and long-term dependencies from the text. However, the effect of this on the performance of DistilBERT does not necessarily need to be very large. Further, the data analysis and LIME implementation provide more valuable results when the data is cleaned. It could be an interesting direction for future research to compare the performance of DistilBERT using text that was not cleaned to the performance of DistilBERT using text that was very thoroughly cleaned. Another option could be to mask the stop words instead of removing them from the dataset, so that the positional encodings would retain the notion that certain words are before or after each other. This is an additional interesting direction for future research.

Since the LIME implementation was only done for articles written by ChatGPT, and those articles were notably different from the articles in the original dataset, it could be valuable to also apply LIME to the articles from the original dataset. The articles in the original dataset were often much longer than the articles written by ChatGPT, so it could be interesting to analyze whether for instance the position of the word in the text affects the feature importance. Further, since most articles were classified correctly in the original dataset, analyzing the articles that were wrongly classified could also provide new insights.

The results of this research provide several implications for fake news detection in organizations. The results are especially relevant for smaller news sites and forums, since they most likely do not have the means in terms of computational power and datapoints to implement a large and complex model such as BERT. This research has shown that DistilBERT, especially with a smaller sequence length, is

an accessible model to implement. Especially in combination with the LIME explanations, the model proposed in this research can be very valuable. A potential practical implementation could be flagging articles that are likely to be fake, and even showing readers which parts of the article determine that the article is most likely to be fake. Such an implementation could potentially prevent the spread of fake news by users that would normally not pay attention to the veracity of the article before sharing. This suggestion is in line with the findings of Pennycook et al. (2021), who stated that after asking the participants in their study to assess the veracity of the article before sharing, the sharing of fake news headlines decreased by 50%. By flagging articles and even highlighting parts of the article that indicate that the article could be fake, users can be nudged to check the veracity of the articles that they read or might share. It is necessary to test this hypothesis further to check whether the spread of fake news actually decreases after flagging articles and highlighting relevant parts in the articles. To test this, a natural experiment could be conducted where a news site or forum implements this feature for some users and not for others, and deliberately create several fake news articles that are not disruptive to society. If the users with the extra feature share less fake news articles than the users without the feature, this could indicate that the feature has the desired effect. Another implementation of this methodology could be in the form of an extension that, when activated, reads the text on a website and provides the LIME output to the user. This way, the validity of information across webpages can be checked easily.

Bibliography

1. Ahmed, H., Traore, I., & Saad, S. (2018). Detecting opinion spams and fake news using text classification. *Security and Privacy*, *1*(1), e9.

2. Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of economic perspectives*, *31*(2), 211-36.

3. Apuke, O. D., & Omar, B. (2021). Fake news and COVID-19: modelling the predictors of fake news sharing among social media users. *Telematics and Informatics*, *56*, 101475.

4. Ashforth, B. E., & Mael, F. (1989). Social identity theory and the organization. *Academy of management review*, *14*(1), 20-39.

5. Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).

6. Biyani, P., Tsioutsiouliklis, K., & Blackmer, J. (2016, February). " 8 amazing secrets for getting more clicks": detecting clickbaits in news streams using article informality. In *Thirtieth AAAI conference on artificial intelligence*.

7. Boehm, L. E. (1994). The validity effect: A search for mediating variables. *Personality and Social Psychology Bulletin*, *20*(3), 285-293.

8. Bronstein, M. V., Pennycook, G., Bear, A., Rand, D. G., & Cannon, T. D. (2019). Belief in fake news is associated with delusionality, dogmatism, religious fundamentalism, and reduced analytic thinking. *Journal of applied research in memory and cognition*, *8*(1), 108-117.

9. Buciluǎ, C., Caruana, R., & Niculescu-Mizil, A. (2006, August). Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 535-541).

10. Castillo, C., Mendoza, M., & Poblete, B. (2011, March). Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web* (pp. 675-684).

11. Chu, Z., Gianvecchio, S., Wang, H., & Jajodia, S. (2012). Detecting automation of twitter accounts: Are you a human, bot, or cyborg?. *IEEE Transactions on dependable and secure computing*, *9*(6), 811-824.

12. Cortes, C., Mohri, M., & Rostamizadeh, A. (2012). L2 regularization for learning kernels. *arXiv preprint arXiv:1205.2653.*

13. David Dunning, Dale W. Griffin, James D. Milojkovic, and Lee Ross. 1990. The overconfidence effect in social prediction. *Journal of Personality and Social Psychology* 58, 4 (1990), 56.

14. DePaulo, B. M., Lindsay, J. J., Malone, B. E., Muhlenbruck, L., Charlton, K., & Cooper, H. (2003). Cues to deception. *Psychological bulletin*, *129*(1), 74.

15. Deutsch, M., & Gerard, H. B. (1955). A study of normative and informational social influences upon individual judgment. *The journal of abnormal and social psychology*, *51*(3), 629.

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

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

18. Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., & Ferrer, C. C. (2020). The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*.

19. Dongare, A. D., Kharde, R. R., & Kachare, A. D. (2012). Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology (IJEIT),* 2(1), 189-194.

20. Effron, D. A., & Raj, M. (2020). Misinformation and morality: Encountering fake-news headlines makes them seem less unethical to publish and share. *Psychological science*, *31*(1), 75-87.

21. Facebook. (2020). *Here's how we're using AI to help detect misinformation*. Retrieved 11 April, from: https://ai.facebook.com/blog/heres-how-were-using-ai-to-help-detect-misinformation/

22. Ferrara, E., Varol, O., Davis, C., Menczer, F., & Flammini, A. (2016). The rise of social bots. *Communications of the ACM*, 59(7), 96-104.

23. Freedman, J. L., & Sears, D. O. (1965). Selective exposure. In *Advances in experimental social psychology* (Vol. 2, pp. 57-97). Academic Press.

24. Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531.*

25. Horne, B. D., & Adali, S. (2017, May). This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In *Eleventh international AAAI conference on web and social media*.

26. Huh, M., Liu, A., Owens, A., & Efros, A. A. (2018). Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 101-117).

27. Jamieson, K. H., & Cappella, J. N. (2008). *Echo chamber: Rush Limbaugh and the conservative media establishment*. Oxford University Press.

28. Johnson, M. K., & Raye, C. L. (1981). Reality monitoring. *Psychological review*, *88*(1), 67.

29. Johnson, M. K., Hashtroudi, S., & Lindsay, D. S. (1993). Source monitoring. *Psychological bulletin*, *114*(1), 3.

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

31. Kaliyar, R. K., Goswami, A., & Narang, P. (2021). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia tools and applications*, *80*(8), 11765-11788.

32. Koppel, M., Schler, J., & Bonchek-Dokow, E. (2007). Measuring Differentiability: Unmasking Pseudonymous Authors. *Journal of Machine Learning Research*, *8*(6).

33. Kumar, S., & Shah, N. (2018). False information on web and social media: A survey. *arXiv preprint arXiv:1804.08559*.

34. Kumar, S., West, R., & Leskovec, J. (2016, April). Disinformation on the web: Impact, characteristics, and detection of wikipedia hoaxes. In *Proceedings of the 25th international conference on World Wide Web* (pp. 591-602).

35. Leibenstein, H. (1950). Bandwagon, snob, and Veblen effects in the theory of consumers' demand. *The quarterly journal of economics*, *64*(2), 183-207.

36. Liu, Y., & Wu, Y. F. (2018, April). Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 32, No. 1).

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

38. Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B. J., Wong, K. F., & Cha, M. (2016). Detecting rumors from microblogs with recurrent neural networks.

39. Meel, P., & Vishwakarma, D. K. (2020). 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.

40. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781.*

41. Mishra, S., Sturm, B. L., & Dixon, S. (2017). Local Interpretable Model-Agnostic Explanations for Music Content Analysis. In *ISMIR* (pp. 537-543).

42. Nasir, J. A., Khan, O. S., & Varlamis, I. (2021). Fake news detection: A hybrid CNN-RNN based deep learning approach. *International Journal of Information Management Data Insights*, *1*(1), 100007.

43. Nickerson, R. S. (1998). Confirmation bias: A ubiquitous phenomenon in many guises. *Review of general psychology*, *2*(2), 175-220.

44. Noble, S. U. (2018). Algorithms of oppression. *In Algorithms of oppression*. New York University Press.

45. Nyhan, B., & Reifler, J. (2010). When corrections fail: The persistence of political misperceptions. *Political Behavior*, *32*(2), 303-330.

46. OpenAI. (n.d.). ChatGPT. Retrieved April 8, 2023, from https://openai.com/blog/chatting-with-ai/

47. Osmundsen, M., Bor, A., Vahlstrup, P. B., Bechmann, A., & Petersen, M. B. (2021). Partisan polarization is the primary psychological motivation behind political fake news sharing on Twitter. *American Political Science Review*, 115(3), 999-1015.

48. Ott, M., Choi, Y., Cardie, C., & Hancock, J. T. (2011). Finding deceptive opinion spam by any stretch of the imagination. *arXiv preprint arXiv:1107.4557*.

49. Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, *22*(10), 1345-1359.

50. Park, M. Y., & Hastie, T. (2007). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society*: Series B (Statistical Methodology), 69(4), 659-677.

51. Pennycook, G., & Rand, D. G. (2020). Who falls for fake news? The roles of bullshit receptivity, overclaiming, familiarity, and analytic thinking. *Journal of personality*, *88*(2), 185-200.

52. Pennycook, G., Epstein, Z., Mosleh, M., Arechar, A. A., Eckles, D., & Rand, D. G. (2021). Shifting attention to accuracy can reduce misinformation online. *Nature*, *592*(7855), 590-595.

53. Pennycook, G., McPhetres, J., Zhang, Y., Lu, J. G., & Rand, D. G. (2020). Fighting COVID-19 misinformation on social media: Experimental evidence for a scalable accuracy-nudge intervention. *Psychological science*, *31*(7), 770-780.

54. Pérez-Rosas, V., Kleinberg, B., Lefevre, A., & Mihalcea, R. (2017). Automatic detection of fake news. *arXiv preprint arXiv:1708.07104*.

55. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1, 2227–2237.

56. Peters, M. E., Ruder, S., & Smith, N. A. (2019). To tune or not to tune? adapting pretrained representations to diverse tasks. *arXiv preprint arXiv:1903.05987*.

57. Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., & Stein, B. (2017). A stylometric inquiry into hyperpartisan and fake news. *arXiv preprint arXiv:1702.05638*.

58. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training.

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

60. Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., & Choi, Y. (2017, September). Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2931-2937).

61. Rayson, P., Wilson, A., & Leech, G. (2002). Grammatical word class variation within the British National Corpus sampler. In *New frontiers of corpus research* (pp. 295-306). Brill.

62. Recasens, M., Danescu-Niculescu-Mizil, C., & Jurafsky, D. (2013, August). Linguistic models for analyzing and detecting biased language. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1650-1659).

63. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " 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* (pp. 1135-1144).

64. Ross, L., & Ward, A. (1996). Naive realism in everyday life: Implications for social conflict and misunderstanding. *Values and knowledge*, *103*, 135.

65. Rubin, V. L., Chen, Y., & Conroy, N. K. (2015). Deception detection for news: three types of fakes. *Proceedings of the Association for Information Science and Technology*, *52*(1), 1-4.

66. Rubin, V. L., Conroy, N., Chen, Y., & Cornwell, S. (2016, June). Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the second workshop on computational approaches to deception detection* (pp. 7-17).

67. Ruchansky, N., Seo, S., & Liu, Y. (2017, November). Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (pp. 797-806).

68. Ruchansky, N., Seo, S., & Liu, Y. (2017, November). Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (pp. 797-806).

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

70. Shahid, W., Li, Y., Staples, D., Amin, G., Hakak, S., & Ghorbani, A. (2022). Are You a Cyborg, Bot or Human?—A Survey on Detecting Fake News Spreaders. *IEEE Access*, *10*, 27069-27083.

71. Shao, C., Ciampaglia, G. L., Varol, O., Yang, K. C., Flammini, A., & Menczer, F. (2018). The spread of low-credibility content by social bots. *Nature communications*, *9*(1), 1-9.

72. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2020). Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big data*, *8*(3), 171-188.

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

74. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929-1958.

75. Subramanian, S. (2017). "Inside the Macedonian Fake-News Complex", *Wired*.

76. Sudipta Basu. 1997. The conservatism principle and the asymmetric timeliness of earnings. *Journal of Accounting and Economics* 24, 1 (1997), 3–37.

77. Szczepański, M., Pawlicki, M., Kozik, R., & Choraś, M. (2021). New explainability method for BERT-based model in fake news detection. *Scientific Reports*, *11*(1), 1-13.

78. Tacchini, E., Ballarin, G., Della Vedova, M. L., Moret, S., & De Alfaro, L. (2017). Some like it hoax: Automated fake news detection in social networks. *arXiv preprint arXiv:1704.07506*.

79. Undeutsch, U. (1967). Beurteilung der glaubhaftigkeit von aussagen. *Handbuch der psychologie*, *11*, 26-181.

80. Van Bavel, J. J., & Pereira, A. (2018). The partisan brain: An identity-based model of political belief. *Trends in cognitive sciences*, *22*(3), 213-224.

81. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L. and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 6000–6010.

82. Vrij, A. (2000). Detecting lies and deceit: The psychology of lying and implications for professional practice. *Wiley*.

83. Vrij, A. 2008. Detecting lies and deceit: Pitfalls and opportunities. *Wiley-Interscience.*

84. Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., ... & Gao, J. (2018, July). Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining* (pp. 849-857).

85. Yao, Y., Viswanath, B., Cryan, J., Zheng, H., & Zhao, B. Y. (2017, October). Automated crowdturfing attacks and defenses in online review systems. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security* (pp. 1143-1158).

86. Zellers, R., Holtzman, A., Rashkin, H., Bisk, Y., Farhadi, A., Roesner, F., & Choi, Y. (2019). Defending against neural fake news. *Advances in neural information processing systems*, *32*.

87. Zhang, X., & Ghorbani, A. A. (2020). An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, *57*(2), 102025.

88. Zhou, P., Han, X., Morariu, V. I., & Davis, L. S. (2018). Learning rich features for image manipulation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1053-1061).

89. Zhou, X., & Zafarani, R. (2018). Fake news: A survey of research, detection methods, and opportunities. *arXiv preprint arXiv:1812.00315*, *2*.

90. Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision* (pp. 19-27).

91. Zuckerman, M., DePaulo, B. M., & Rosenthal, R. (1981). Verbal and nonverbal communication of deception. In *Advances in experimental social psychology* (Vol. 14, pp. 1-59). Academic Press.