Using Neural Networks for Misinformation Classifications in English Textual Documents

Student Name: Isi Ali
Supervisor Name: Shauna Concannon
Submitted as part of the degree of BSc Computer Science to the
Board of Examiners in the Department of Computer Sciences, Durham University

Abstract—Context: With the increased rate in which news, articles and media is spread, the rates at which false information is able to be consumed is a huge problem going forward in the 21st century with potentially catastrophic consequences. The sheer magnitude of content disseminated daily makes it impractical for human efforts alone to effectively address the issue of misinformation. Consequently, the deployment of AI-driven tools becomes crucial in initiating a robust response to tackle the spread of misinformation.
Aims: To outline and affirm reported traits of conspiracy, both graphically and statistically, as well as advance already present neural network models that exist with the aim of classifying articles as either conspiracy or mainstream.
Method: We take the already present work done in evaluations of the LOCO dataset and expand the analysis of conspiracy traits to represent a much wider corpus of the LOCO dataframe, as well as using a hybrid Convolutional Neural Network along with advanced NLP techniques to improve contextual understanding of the model.
Results: All reported traits of misinformation are confirmed across a large sub-sample of each corpus of text. Ability to accurately predict whether an article is conspiracy or mainstream is high consistently above 80%, however more work could be done to achieve higher rates of successful classification.
Conclusion: The biggest takeaway from this paper is the objective differences discovered between conspiracy and mainstream articles, meaning it is possible to accurately identify texts which contain misinformation with the intent to cause harm using Artificial Intelligence tools. Further experimentation and research into these methods is needed and the current proposed models can be improved upon.

Index Terms—Deep Learning, Natural Language Processing, Misinformation Detection, Text Classification

1 Introduction

The advent of social media and freedom to express information disguised as truths to the masses has given rise to what many social scientists and linguistic experts say is a huge problem that needs to be addressed [1]. Coupled with the recent advancements of Large Language Models (LLMs), and their ability to produce large quantities of text, the problem is likely to become exacerbated and reach unprecedented scales as society adopts Artificial Intelligence in our writing and research [2].
Attempting to remedy this by policing GPT usage is near impossible, and so the next call of action would be to look to AI itself to assist in the detection of misinformation. The key word here is to assist, and not replace human fact checkers. The complexity and subtlety of language results in misinformation having the ability embed itself into textual documents, but the sheer volume of text released does call for changes to be made in its detection.
How to quantify the characteristics of an article to classify it as 'conspiracy' is one of the questions social scientists are always on the search for answers to [3]. The work done by Mompelat et al., [4] provides a very solid foundation for metrics to begin the process of assisting humans in determining misinformation in English text where multiple different traits were listed as being attributes of conspiracy text. The significance of this work would be to make the daunting task of monitoring information online, and attempting to combat the spread of misinformation, a do-able task [5]. Currently the task is almost impossible, reported so as early back as 2008 [6], with the advent of the 24-hour news cycle. Not to mention the widespread accessibility to platforms such as X (formerly known as Twitter) and TikTok allowing unprecedented amounts of people to have their words reach millions of people at just the tap of a button [7].
A conspiracy theory is defined as a made up belief acting covertly to achieve some malevolent end, with an orchestrated plan to achieve a secret harmful goal [4].
This definition highlights that any proposed model made to flag articles containing misinformation should focus most on articles of misinformation that specifically have the intent to cause harm, since it would be impossible to develop the deep learning architecture to deal with the entirety of text that is released, due to the uncontrollable speed of the spread of information online.

1.1 Sociological research into misinformation

A large proportion of sociology studies that relate to misinformation focus on fake news. For example, one paper [8] provides a comprehensive review of the phenomenon of fake news, defining it as false information designed to

deceive, that mimics the look of real news. This aligns with the value of a conspiracy theory our model will uphold with the aim to detect articles that are intending to cause harm.
The paper [8] notes the notoriety of the problem of misinformation spreading, attesting heavy emphasis that this reality we face worsened during and since the 2016 U.S. presidential elections. Although research suggests that the direct audiences are relatively small, they are highly engaged. However, it also noted that the impacts can be substantial despite the sizes of audiences as they are likely to amplifying the spread of second-hand disinformation.
'First impression bias' is a term used by behavioural scientists, to mean a limitation in human information processing that causes us to make quick and incomplete observations about others based on the first perceptions [9]. Given the fast-paced nature of social media platforms like TikTok, where users consume vast amounts of information in brief periods, it is increasingly crucial to ensure the accuracy of information shared in articles to prevent the spread of misinformation.
The paper also details the importance of tackling the problem with more direct approaches, and how simple 'fact-checking' algorithms are insufficient for tackling the mammoth tasks they face. The 'backfire effect' [10], where people report believing even more in misinformation after they have seen an evidence-based correction aiming to rectify it, testifies to this.
Further sociological research even calls for moving away from the term 'fake news' due to its ambiguous nature, instead differentiating between various types of misleading information: those spread with the intent to deceive, versus honest mistakes without the intent to harm [11]. This paper details the threat of the hybrid news system where traditional and new media actors operate according to overlapping and competing media logic, blurring the line between professional journalism and user-generated content, complicating the propagation of information.
These two papers confirm the dangers of misinformation and show that work outside of simple fact-checking must be done to detect and strike down articles promoting misinformation before people have the chance to see them. Looking towards neural networks to assist in this problem is a very probable solution with much research ongoing in the field to achieve such a task.

1.2 Contributions

This paper will detail two primary contributions to the field of misinformation detection. The **first** will be to run a large-scale comparative study between articles which carry the label of conspiracy and articles carrying the label of mainstream, verifying the features characteristic of conspiracy theories established via a qualitative analysis of a small sample of data from the LOCO corpus identified by [4]. The dataset that will be utilised has two core dataframes of which will be delved into later in this paper. One contains the raw text and baseline features such as number of words, sentences and social media shares, and another with latent features extracted from two NLP pythonic modules:

1) LIWC analysis [12] - The Linguistic Inquiry and Word Count is a text analysis software that quantifies various aspects of language including emotional, cognitive and structural components. This is achieved by counting words in psychologically meaningful categories.
2) Empath analysis [13] - The Empath module is a tool that analyses text to identify and categorise words and phases across a wide range of pre-defined categories similar to LIWC.

The advantage of having both is that the LIWC analysis will be able to affirm reported traits of conspiracy with data visualisation and analytic tasks, whilst the Empath module is open-source, allowing further research on alternative datasets that may not have the latent features extracted without explicitly purchasing a LIWC license.
As this paper will go on to detail, it was found that the traits reported by the linguistic breakdown of the LOCO corpus are in fact correct, backed up with tests, both statistical and graphical with samples of 20,000 articles of each subcorpus. This lays out definitively what traits of text are frequently attributed to conspiracy documents. justifying this usage is essential for inserting features into a deep learning architecture to produce a predictive model classifying texts as to whether they contain misinformation - a trait of conspiracy.
The **second** contribution to the field of misinformation detection is advancing neural network models, most significantly by incorporating latent features into the model, providing deeper insights into textual understanding for the model. This is achieved through the usage of additional NLP techniques such as word embedding analysis and EMPATH computation of the texts.

1.3 Deliverables

Fundamentally, this project aims to create a deep learning model able to classify a series of documents by their word embedding and specific Empath features. We will train two distinct models, one with LIWC features embedded into the training process, and one without. The one without will be the significant model as it will allow reproducibility and wide range usability on datasets without the requirement of a LIWC license.
In addition, statistical work details how conspiracy documents tend to ask more questions, mention other conspiracy theories, more frequently direct text towards the reader through 1st and 2nd pronoun usage amongst many other reported traits. This will be displayed visually with the aid of various plots from histograms, box plots and Kernel Density plots, as well as through the usage of various T-tests showing the significant and objective disparities between conspiracy and mainstream, where in every instance a T-test was ran, the null hypothesis was rejected.
Finally this paper will detail the results of all models trained on Empath and word embedding features perform on external datasets, such as the Fake News Dataset from Kaggle. The three models that will be deployed for this external testing will be:

1) Replica of the FND network based on the architecture from Kaliyar et al,. [14] seeing if the 98.36% accuracy rate can be replicated.

2) The FND architecture with enhanced NLP techniques ensuring the model can capture the semantic and contextual levels of the textual inputs.
3) A RNN-CNN hybrid model for enhanced feature extraction from the word embeddings and Empath features, integrating the merits of extracting different aspects of linguistic information, thus strengthening the semantic understanding of the presented texts. This will be based on the work done byNasir et al. [15].

2 Related Work

Deep Learning (DL) has been extensively applied to the field of misinformation detection [16]. To place our paper into the wider context of the field we review literature related to the application of DL in detecting misinformation in English text. There are a variety of techniques used, from fact-checking algorithms that cross reference news content against trusted sources and databases, to network analysis [17], breaking down patterns of news on social media and the internet and looking at the spread of nefarious content to predict its existence [18].

To structure this discussion of existing solutions, we partition the existing literature into Deep Learning methodologies, NLP techniques applied to the field, and ethical and sociological research examining the applications of such algorithms.

DL has become an emerging technology among the research community and has proven to be more effective in recognising fake news than more traditional Machine Learning methods. DL has some particular advantages over machine learning:

1) Automated feature extraction.
2) Light dependence on data pre-processing
3) Improved accuracy for text classifications. [19].

Following on from this, programming frameworks such as PyTorch have boosted the usage and robustness of DL-based approaches. Multiple DL methodologies and approaches have been implemented from diffusion networks [20] [21], association networks [22] and propagation networks [23].

When working with textual data, the DL approach is heavily aided by techniques from the field of Natural Language Processing (NLP). This is a field of artificial intelligence with the ultimate objective of enabling computers to understand, interpret and analyse human language in a valuable way. NLP involves applying algorithms to convert unstructured language data into a form that computers can understand. Natural Language Understanding (NLU) is the subset of NLP that is most relevant for misinformation detection [24]. The goal of NLU is to comprehend the nuances of language, including context, intent and sentiment beyond just recognising words and sentences.

2.1 Introducing the LOCO dataset

The first step in utilising AI tools for such a task requires a dataset that is adequately labelled. The primary dataset we will use in this project is 'LOCO: The 88-million-word language of conspiracy corpus' [25], which we will refer to as the **LOCO dataset**. This dataset is a collation of articles that have been classified into two groups: 'conspiracy' and 'mainstream'. There are 23,397 'conspiracy' documents and 72,806 'mainstream' documents. This dataset contains an enormous quantity of metadata relating to linguistic features, as well as publication date, measures of social media engagement, size, traffic and political bias.

The dataset is made up of 37 seeds (categories) in total, 20 being collated from popular conspiracy theories (such as Flat Earth theory, 5G hysteria and provably false information surrounding Bill Clinton) and 17 seeds from the paper Douglas and Sutton et al., [26] which analysed a further set of conspiracy theories that didn't necesarilly hit the mainstream, but were still believed by the many.

An external review done by Mompelat et al. [4] concludes that 85% of conspiracy theories are labelled correctly, and 92% of mainstream documents are labelled correctly. Features of conspiracy theory articles are also noted by the paper and they are listed in Table 1. These features form the basis of what information is to be fed into the deep learning model.

For the purposes of this project, the labelling of the articles denoted within the LOCO dataset will be used as a ground truth. This is in spite of the existence of 15% of conspiracy and 8% of mainstream documents labelled incorrectly. This should not pose too heavy of an issue as this model would not be producing objective outputs regarding textual documents, but instead providing guidance for human fact-checkers as to which articles to direct their attention to.

TABLE 1
Features of Conspiracy

Explicitly fact-checking beyond inconsistencies	
Syntactic clues	• Paraphrasing instead of quoting • Frequent questions towards the reader • Atypically named entities
Semantic clues	• Excessive usage of capitalisation • Unconventional usage of punctuation • Usage of the 1st and 2nd person
Mentioning other conspiracy theories	
Sensationalism used to excite emotion	
Having a contradictory opinion to mainstream media	

These traits were identified from a review of 40 articles labelled conspiracy and 40 articles labelled mainstream. This leaves the scope open for the further analysis, affirming whether the traits listed in Table 1 hold true for the entire dataset.

2.2 Natural Language Processing Tools (NLP)

There are a variety of NLP tools and techniques aimed at identifying inaccurate, misleading or false information. Below is a brief overview of some NLP tools and approaches that have been applied in this field.

1) **Fact-checking algorithms**: These are specialised NLP systems designed to verify the authenticity of statements within a text by cross-referencing them with reliable data sources and databases. Such algorithms are often paired with Named Entity Recognition [27](NER) to identify proper nouns and factual claims for validation. NER tools identify and categorise key entities within the text such as names of people, organisations and locations to enhance the cross-referencing capabilities providing the quick identification of false claims. However, research does point to such algorithms being inadequate when deployed on their own [28].
2) **Sentiment analysis**: Misinformation often carries a strong emotional charge to influence readers. Tools which can apply quantifiable metrics to be inferred by a machine can help identify signals in texts that may indicate biased or manipulative content. There are various tools to achieve this, most notably the VADER (Valence Aware Dictionary and sEntiment Reasoner) algorithm , a simple rule-based model for general sentiment [29]. It takes a piece of text and returns a dictionary containing sentiment scores for the text with 4 keys: Negative, Neutral, Positive and a compound being the overall score between -1 and 1.
3) **Word Embeddings**: Such a tool creates vector representations of words in a high-dimensional space, capturing semantic similarities based on the context in which words appear. Misinformation detection models can use these embeddings to understand context and detect subtle manipulations or inconsistencies in text.
4) **Contextual Embeddings**: These are another type of word representation that captures the meaning of a word in relation to the words around it, effectively understanding the context in which a word is used. Unlike word embeddings, which generate a single static representation for each word regardless of its context, contextual embeddings provide dynamic word representations. These representations capture nuances in meaning based on the use of the words in a sentence. Bidirectional Encoder Representation from Transformers (BERT) and its variants are tools that aid in detecting more sophisticated misinformation that relies on nuanced or ambiguous statements.

2.2.1 Word Embeddings

A key aspect of NLP for any task in computer science is being able to convert raw text into a numerical format. These are the word embedding tools mentioned above. There are two primary techniques that are used, both having proven application in the field of misinformation detection: GloVe (Global Vectors) and Word2Vec (Word to Vectors).
GloVe is an unsupervised learning model that is based on ratios of probabilities from the word-word co-occurrence matrix. The model is designed to aggregate global-word co-occurrence statistics from a corpus, and then these statistics are used to directly inform the learned word vectors. The primary insight of GloVe is to leverage the overall statistics of word occurrences in a corpus to produce word embeddings.

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij} \right)^2 \quad (1)$$

Equation 1 defines the loss function **J** that the GloVe model aims to minimize during training. The goal is to learn the word vectors in such a way that their dot product equals the logarithm of the words' probability of co-occurrence.
The other notable technique for creating word embeddings of textual data is Word2Vec [30], developed prior to GloVe in 2013 by Google. Word2Vec employs a three-layer neural network (input, hidden and output) to perform centre context word-pair classification tasks where word vectors are the by-products, with two variants to the model:

1) Continuous bag-of-words (CBOW) - Predicts a target word based on its context words, taking in the context words as input and trying to predict the target word in the centre of the context.
2) Skip-gram - Does the opposite of CBOW, using the target word to predict the surrounding context words.

Word2Vec is a predictive model and a feed forward neural system that learns vectors to improve the predictive capacity. Similar words are located together on the vector space and arithmetic operations on word vectors can pose semantic or syntactic relationships. Such a technique has been deployed for the specific task of misinformation detection to varying success, such as in the work done by Mallik et al. [31].

2.3 Neural-Networks

There are also multiple other deep learning methods which will be combined in the eventual model deployed to the LOCO dataset. What is first necessary to understand is the role of convolutional neural networks when it comes to neural networks and their application in this field. The mathematical depiction of a convolutional neural network is depicted below [32].

$$C(i,j) = \sum_{c=0}^{C-1} \sum_{m=0}^{K-1} \sum_{n=0}^{K-1} I_c(i + m \cdot S, j + n \cdot S) \cdot F_c(m,n) \quad (2)$$

In the context of misinformation detection using Convolutional Neural Networks (CNNs), the input feature map I typically represents encoded textual information. This encoding might be in the form of word embeddings or character-level representations of size H *x* W, where H might correspond to the number of tokens in a sequence (like words or characters), and W to the dimensionality of the embeddings. A convolutional filter F, of size K *x* K, is then applied to these textual representations to produce an output feature map. This map is generated through a sum of element-wise multiplications and additions over the input map and the filter, effectively capturing local and contextual patterns within the textual data.
These local patterns may include n-gram features of character combinations that are indicative of misinformation, such as specific phrases, stylistic patterns or analytic anomalies. By sliding the convolutional filter across the embedded text - considering the stride and possible padding to maintain

dimensionality - the CNN is able to detect these local features and learn which patterns are most indicative to an article containing misinformation.
CNNs are not the only type of neural networks that have applications in this field. Recurrent Neural Networks (RNNs) are another type with heavy literature and research showing promising results in the pursuit of detecting the truthfulness of an article. These neural networks process sequences by iterating through the sequence elements and maintaining a state that contains information relative to what it has seen so far. The basic formula for a recurrent neural network is as follows:

$$h_t = \text{activation}(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \tag{3}$$

$$y_t = W_{hy}h_t + b_y \tag{4}$$

These two equations are the foundation of RNNs and they are particularly pertinent in this specific domain of misinformation detection. RNNs are adept at processing sequential data, making them suitable for analysing textual content, treating text as a sequence of words. The unique characteristic of the hidden state *ht*, updated at each time step of equation (3) effectively serves as a memory of the network that captures information from all previously seen elements in the sequence.
The final key in neural network research in this field is the existence of hybrid models. CNN-RNN hybrid models have been extensively explored for text classification [33] with examples existing for fake news detection in social media [34]. Combining these into a single model harnesses the strengths of both architectures. CNNs excel at extracting salient local features through their hierarchical processing of input data, identifying underlying patterns such as keyword clusters or semantic groupings within a text. On the other hand, RNNs are adept at analysing the temporal sequence of data, capturing the context and flow of information over longer spans of text. This is essential for understanding nuanced narratives with complex argumentation which is common in appearances of deception which can emerge in articles without being caught before publication.

2.4 FNDNet

A number of studies have used neural networks for the identification of misinformation in text.

1) Ahmed et al. performed fake news classification using linear regression based on uni-gram models achieving an accuracy rate of 89% [35].
2) Ahmed et al. further improved on work in this field by using a linear support vector machine, achieving an accurate rate of 92% [35].
3) O'brien et al. utilised convolutional neural networks and a form of semantic analysis, focusing on the sensitivity of texts present in the corpus, with this approach achieving an accuracy of 92.10% [36].
4) Ghanem et al. incorporated word embeddings and n-gram features to detect the stance in fake news and achieved an accuracy rate of 48.80% [37]
5) Singh et al. used a combination of support vector machines as a classifier with LIWC textual analysis for latent features and achieved an accuracy of 87% [38].

The primary piece of work produced in this field which reports the best results is that produced by Kaliyar et al. [14]. This implementation of misinformation detection has a reported accuracy rate of 98.36% indicating state of the art results. The FakeNewsDetector network (FNDNET) was designed to learn 'discriminatory' features for fake news classification through multiple hidden layers built into a deep neural network. The network can be seen visually in fig1.

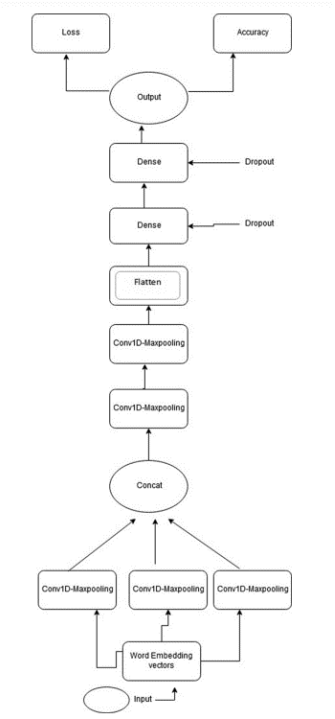

Fig. 1. Computational graph for the FNDNet model.

The architecture of this network consists of 4 key features. Firstly, the textual input is run through NLP methodologies for feature extraction, specifically the Word Embedding vector GloVe [39].
The preference for a GloVe word embedding to be deployed into the neural network is the ability of parallel implementation which Word2Vec does not support. This makes it easier to train the model over large datasets. In addition, the GloVe algorithm combines the benefits of Word2Vec skip- gram model in word analogy tasks with matric factorization using global statistic information.
The second portion of this neural network consists of a non-sequential run of 3 convolutional neural networks being one dimensional in nature due to this architecture dealing with textual data. These layers are then concatenated into a sequential, convolutional neural network, running the data and calculated word embeddings through two further layers, before being flattened to convert the numerical data into processable information for the system to analyse. These densely connected layers can give rise to overfitting - where the model learns the training data too well, reducing

the generalisability of the model when tested on unseen data [40]. To circumvent this problem, [14] incorporated dropout layers, discarding some of the data randomly during the training process. This method, known as dropout, essentially 'turns off' a random subset of neurons at each training step, preventing them from co-adapting too closely to the training data and thus encouraging the network to develop more robust features that are useful in conjunction with many different random subsets of the other neurons [41]. Dropout has been shown to significantly improve the performance of neural networks on supervised learning tasks in vision, speech recognition, computational biology and most importantly, document classification [41], making it an effective regularization technique [42].
The data that was used for training was the Fake News Dataset from Kaggle, was then evaluated using a binary class to predict whether the content of a given article was either real or fake with a loss and accuracy metric. The paper reports an accuracy rate of 98.36% , surpassing performance on all other reported models in detecting fake news, hence the decision to use this model as a basis for my misinformation detector.

A combination of word embedding techniques and sentiment analysis will be incorporated into the model produced in this project.

3 METHODOLOGY

3.1 Affirming Conspiracy traits

Before being able to deploy any form of machine or deep learning on a task, developing a deep understanding of the content we are classifying and affirming quantifiable differences between the two groups we are classifying is essential. As per the table laid out in the introduction there were 6 key qualities of conspiracy theories.

- Syntactic clues (specifically, frequent questions)
- Semantic clues
  - Excessive usage of capitalisation
  - Unconventional usage of punctuation
  - Usage of 1st and 2nd person
- Mentioning other conspiracy theories
- Sensationalism used to excite emotion

With research mentioning the inadequacy of fact-checking algorithms alone, and the focus on using these findings to feed information, deploying these algorithms on the articles is not necessary for this project. In addition, these algorithms are able to stand on their own and so can be combined with the neural network to enhance misinformation detection.
The paper that produced the traits of conspiracy [4] were inferred from a sample of 40 documents initially. These document were selected from the LOCO corpus on articles relating to Sandy Hook specifically, with 20 documents from the conspiracy subcorpus, and 20 from the mainstream subcorpus. Then further analysis was performed on 20 of each subcorpus, this time relating to the Coronavirus conspiracy theory.
This relatively small subset of the documents used to infer traits does call for caution when using these traits for classifying a document as either conspiracy or mainstream when there in total 96743 documents present in the LOCO dataset.
These documents were collated from lists of conspiracy and mainstream websites, drawn from Media Bias/Fact Check (MBFC) scores. This is an independent online media outlet that evaluates and rates the bias and factual accuracy of various media sources, including newspapers, news TV channels, and websites. MBFC categorises media sources on a spectrum from extreme left to extreme right, producing a numerical value that gauges the reliability and bias of information presented by media outlets.
Along with URL extraction, text extraction and extensive cleaning through the Python Goose package, the main body of text from each HTML file was obtained. Then metadata such as the document title, language and lexical feature extraction were obtained using analysis tools like LIWC and Empath.
With this background understanding, affirming the reported traits of conspiracy can begin, with various pythonic tools, from visual tools like matplotlib and seaborn to language tools such as 'language_tool_python' which is an open-source grammar tool.

3.1.1 Excessive questions

It is reasonable to assume articles aiming to convince you of a certain viewpoint would prioritise asking questions to the reader encouraging them to question what they may already believe to be true. An article spreading facts would simply lay out the truth as opposed to prodding the audience with rhetorical statements. The number of questions that were present in each article was calculated by counting the number of question marks that appeared. It was decided against counting questions words (e.g., how, why, what, etc.) as this could introduce ambiguity as these words can be used outside of a question, whereas a question mark would objectively depict a question.
To test whether this was a trait that existed outside the sub-samples taken out of the Coronavirus and Sandy Hook subcorpus', a sample of 40,000 documents were taken: 20,000 from each subcorpus, to ensure a balanced representation of each group. Results were substantial, showing that conspiracy articles included **3x as many questions as mainstream articles**, depicted in fig 2.
On top of this, when looking at the sampled articles, within the conspiracy subcorpus, the 5 articles that asked the most questions included 473, 423, 301, 246 and 160 questions respectively. In comparison, within the mainstream subcorpus, the 5 articles that asked the most questions were 166, 108, 89, 85 and 83. This stark contrast leads us to our first attribute that could be fed into a neural network to detect articles that are likely to lean towards conspiracy, and by extension misinformation.

3.1.2 Excessive usage of capitalisation

This was a trait reported that links with sensationalism and overt emotional language directed to the user. The capitalisation ratio was calculated with the number of words with more than one uppercase letter divided by the total number of words in the document. This provides a measure

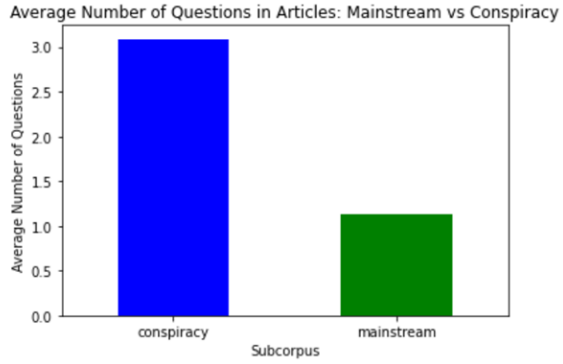

Fig. 2. Bar chart showing average number of questions asked per article.

for how frequently excessive capitalisation was used. This

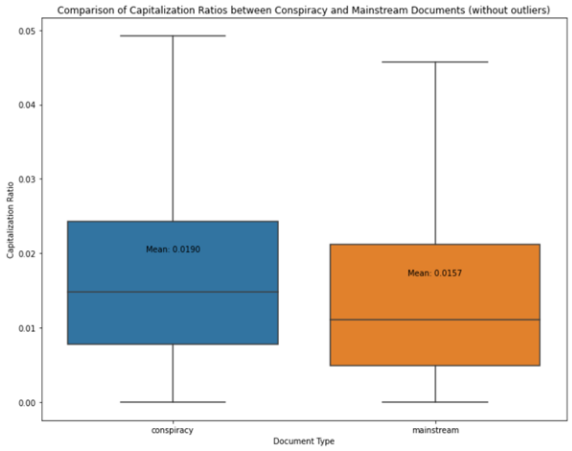

Fig. 3. Box plot comparing distribution between the two subcorpus

box-plot 3 clearly depicts conspiracy articles having a larger capitalisation ratio than mainstream articles. To determine if there is a statistically significant difference, we perform a Mann-Whitney U test, also known as a Wilcoxon rank-sum test. This is a non-parametric test that compares two independent samples to determine a difference in population medians. As the data for the capitalisation ratio is unlikely to be normally distributed, this type of test is optimal for observing a proposed statistical significance here.

$$U = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1 \tag{5}$$

This test ranks the values from both groups together, then calculates the sum of the ranks for the observations from each of the two groups. The U statistic is obtained by evaluating the differences between the sum of the ranks for each group and what would be expected by chance if the two distributions were identical. The null hypothesis for such a test would insinuate the two samples are drawn from the same population and so have little difference between the two groups. n_1 and n_2 represent the sample sizes of each group, and R_1 is the sum of the ranks in the first group. is the sum of the ranks in the first group.
After using Scipy.stats built in formula, a P-value of 2.79×10^{-159} was calculated. This tells us that the difference in capitalisation ratios between conspiracy and mainstream articles is statistically significant.

3.1.3 *Unconventional usage of punctuation*

This was a relatively difficult trait to measure as defining what conventional punctuation is in itself poses various challenges. With such a metric being quite subjective, the initial test was to simply measure the varying punctuation usage between both subcorpus, and seeing if there is any significant differences. To do this another statistical test was deployed. This was calculated by simply counting the number of punctuation marks that appeared utilising the 'string' python module.
Despite textual data being inherently discrete, the ratio of punctuation would transform this data into continuous values that could be analysed for their distribution. Then with the incredibly large population size of our test data being 40,000 textual documents, we were able to make an assumption of a normal distribution of the data. This was supported by the Central Limit Theorem which states that if you have a large enough sample size, the distribution of the sample means will be approximately normally distributed, regardless of the shape of the population distribution from which samples are drawn [43].
With the knowledge that the ratio of punctuation used was both continuous and followed a continuous distribution, a statistical t-test was appropriate for determining if there was meaningful differences between the usages of punctuation in conspiracy and mainstream documents. The t-test is specifically designed to compare the mean of two groups. By calculating the punctuation ratio for each text and then comparing the means of these ratios between the two groups, we are able to directly compare the central tendencies of these groups in how punctuation is used providing stylistic differences between the two groups.

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{6}$$

This formula calculates the t-statistic by taking the difference between the two sample means and dividing it by the standard error of difference between those means. This reflects the variability and size of each sample. This was once again performed with the scipy module and the results were as follows:

- The conspiracy punctuation mean was calculated to be 0.0233. This means that 2.33% of characters on average in conspiracy texts were punctuation marks.
- The mainstream punctuation mean was calculated to be 0.0249. This translated to 2.49% of the characters in mainstream texts being punctuation marks.
- The T-statistic was calculated to be -22.226. A T-statistic far from zero, either positive or negative, suggests a significant difference between the groups. The negative sign here highlights the first group (conspiracy) being having a lower value than the latter (mainstream).
- The P-value was calculated to be 8.87×10^{-109} . This is far smaller than the confidence value, meaning there is an extremely low probability that this observed difference is due to random chance.

3.1.4 What is unconventional?

Affirming an objective difference between the two groups is useful, however this statistical significance does not confirm the trait, as what does constitute as 'conventional'? To commence this we begun by sampling 20,000 instances from each subcorpus category to ensure analysis is based on a representative subset of the data, allowing our findings to be generalisable for the dataset as a whole. Utilising the Python tool 'language_tool_python' for punctuation analysis was essential for this analysis. This is an open-source proof-reading software, allowing for the automated and detailed inspection of punctuation usage within texts. Various rules are embedded into this framework and the three of which were chosen to perform the tests were:

- PUNCTUATION_RULE: General rule for punctuation errors.
- COMMA_PARENTHESIS_WHITESPACE: Specific rule for white-space usage with commas and white-space.
- SENTENCE_WHITESPACE: Rule for the correct usage of white-space in sentences specifically.

Each rule has its own criterion for identifying errors, based on the syntax and punctuation standards of the language. Due to the nature of many of the conspiracy subgroups that exist within this dataset, since they are mostly in America we chose American English . It was found in the subsample taken that conspiracy documents were twice as likely to contain punctuation errors as shown in fig 4. Not only were

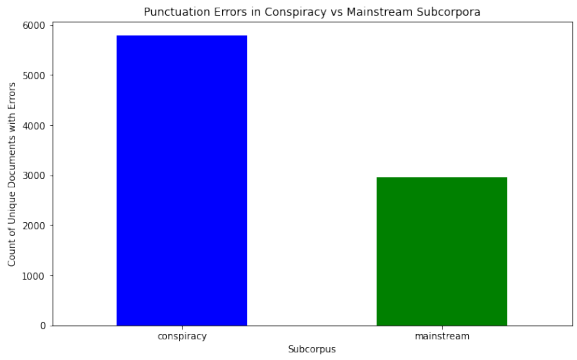

Fig. 4. Bar chart comparing punctuation errors in a subsample of conspiracy and mainstream documents.

errors more common in conspiracy articles in general, the frequency of errors appearing in each individual article was also much larger as depicted in fig 5. Both histograms depict that a high frequency of the documents in each group contain a low number of punctuation errors, peaking around 0-5. In both cases, this quickly tapers off as the number of errors increases. What is significant here is the extension of the respective tails, conspiracy documents have many more instances of documents with over 30 errors, reaching a maximum of 40 errors. In contrast mainstream documents had a sparsity of documents with errors greater than 12, with the tail of this histogram ending at 30 with one article.

3.1.5 Usage of 1st and 2nd person

This was a relatively straightforward trait to confirm, with the process simply involving counting how many times various first or second pronouns appear in the text. It is

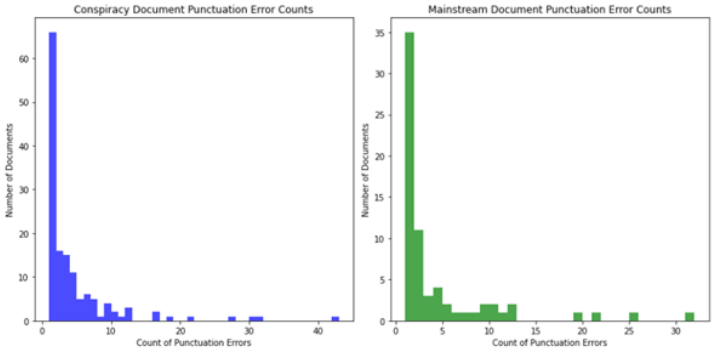

Fig. 5. Histogram comparing frequency of punctuation errors in a subsample of conspiracy and mainstream documents.

logical that conspiracy articles would tend to have a greater amount of such pronouns with an emphasis on connecting to the reader as opposed to sharing facts. Such pronouns open up the discussion for the authors opinion to be shared as opposed to objective truths. The findings from the analysis of subset sample sizes of 20,000 for each subgroup concluded that documents labelled conspiracy contained more usage of these pronouns as depicted by fig 6 for 1st person pronouns, and fig 7 for 2nd person pronouns.

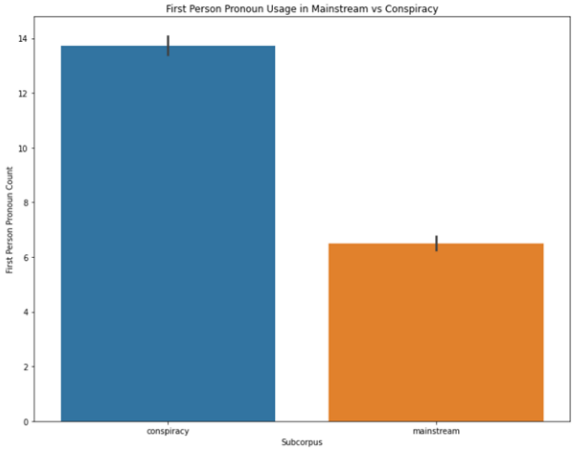

Fig. 6. Bar chart comparing usage of first person language in a subsample of conspiracy and mainstream documents.

In addition to the manual confirmation of this by counting the number of appearances of pronouns like 'I, we, me, you', the LIWC analysis performed by the authors of the LOCO dataset also provides continuous numerical data representing first and second pronoun usage. LIWC analysis specifically looks at the usage of the pronouns 'I', 'we' and 'you'. With the existence of this continuous data, and of such large documents, the central limit theorem could once again be applied. With this T-test statistical tests could once again be deployed, and for all three, a very small P-value was calculated meaning the null hypothesis of no differences between the two categories could be rejected. This confirmed statistically significant differences between the two groups.

This table depicts a greater difference in the usage of 2nd person pronouns between the two subgroups with

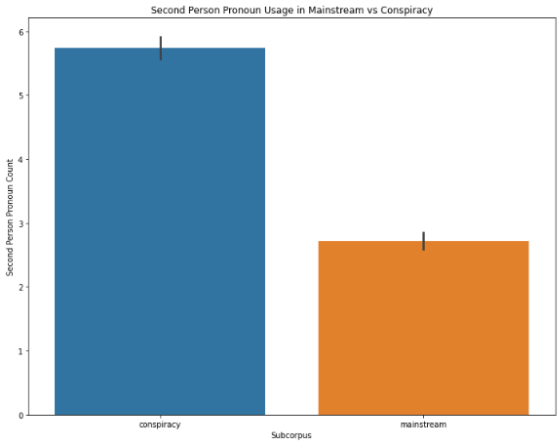

Fig. 7. Bar chart comparing usage of second person language in a subsample of conspiracy and mainstream documents.

	T-test value	P-value
LIWC-I	4.15	3.30×10^{-5}
LIWC-YOU	29.67	1.99×10^{-191}
LIWC-WE	17.48	3.92×10^{-68}

TABLE 2
T-test results summary for pronoun usage.

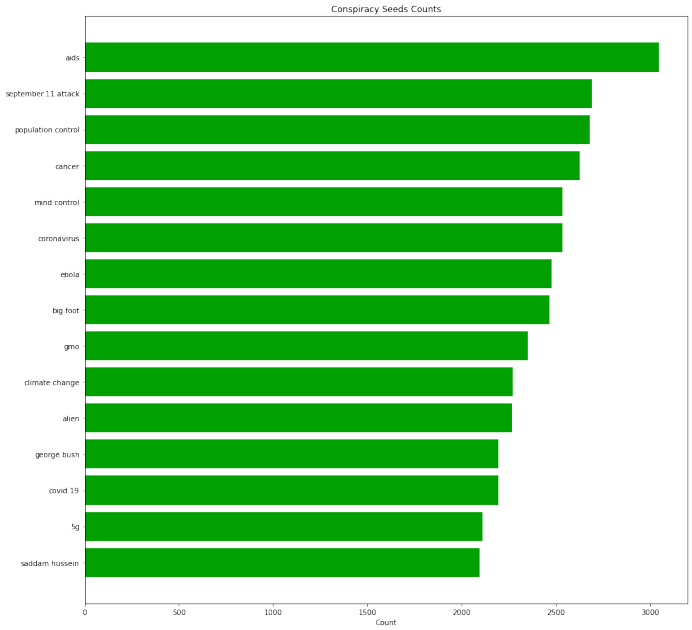

Fig. 8. Bar chart showing the distribution of conspiracy theories present in the LOCO dataset.

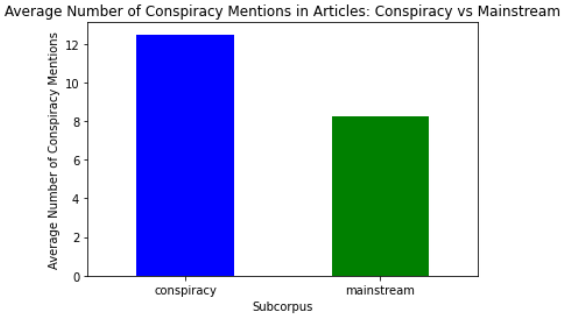

Fig. 9. Bar chart showing the distribution of conspiracy theories present in the LOCO dataset.

conspiracy documents heavily deploying such pronouns more frequently. It is understandable that in order to persuade people of the incorrect information disseminated in conspiracy articles, directly addressing the audience would be most effective .

3.1.6 Mentioning other conspiracy theories

The following trait was again relatively simple to confirm and just required some textual pre-processing before running on the same sample of 40,000 documents. There are a wide variety of conspiracy theories that are depicted within the LOCO dataset, with a graphical representation of the distribution of such conspiracies shown in Fig 8. The logic behind this trait comes from humans having strong tendencies to seek patterns and connections in the world around them. Conspiracy theories often flourish within specific communities that share similar worldviews, and so bringing up other conspiracy theories would lead to confirmation bias for the reader, reinforcing readers' confidence in false information. In line with the previously confirmed traits, conspiracy documents did indeed mention conspiracy theories more often than mainstream ones did. Fig 9 shows this difference with a 50% change in mentions in the conspiracy subcorpus.

3.1.7 Sensationalism to excite emotion

Sensationalism is a method of journalism which focuses on exaggerating, distorting or using sensational aspects of news stories or events to attract readers attention. Complex issues are usually oversimplified and shocking and dramatic aspects of stories are highlighted. To measure this we can look toward the LIWC and Empath features that were calculated by the author with the release of the LOCO dataset. LIWC was developed for analysing the psychological dimensions of language use. The core of LIWC analysis lies in its dictionary, which categorises thousands of words and word stems into various psychologically relevant categories, such as positive emotions, negative emotions, cognitive emotions, social processes, personal pronouns, tenses and many others. By calculating the frequency of these words in these categories within a given text, LIWC provides insights into the writers psychological state, social relationships, thinking style and personal concerns. This makes LIWC an incredibly valuable source of textual analysis, especially when comparing the two subgroups and affirming statistically significant differences between them.

There are over 93 predefined categories so feeding every category into the neural network would result in excessive noise being flooded into the system, impeding the network's ability to accurately infer information about text. Because of this, in through the usage of the official LIWC documentation, the most relevant categories were selected to use in the final network. These were:

- **LIWC_affect**: Measures the overall emotional tone of the text, capturing both positive and negative emotions
- **LIWC_posemo**: Represents positive emotions, indicat-

ing content that may be trying to evoke optimism or happiness in an exaggerated manner.
- **LIWC_negemo**: Represents negative emotions, which are often used in sensationalist content to provoke fear, anger or sadness.
- **LIWC_anx**: Measures the presence of anxiety-related words, which can be a component of sensationalist messaging aimed at provoking stress or worry.
- **LIWC_anger**: Indicates the use of language associated with anger, which can be prevalent in sensational content designed to incite outrage.
- **LIWC_sad**: Reflects the usage of sadness-related words, potentially used to evoke sympathy or emotional distress.
- **LIWC_discrep**: Includes words that express disagreement, denial and divergence from expectations, with expressions of doubt or negation.
- **LIWC_differ**: Encompasses language that signifies distinction, diversity and separation.

The Empath python module is a tool similar in spirit to LIWC but operates with a broader and more flexible approach to analysing text for various psychological, thematic and emotional elements. Similar to LIWC it has many predefined categories designed to facilitate the analysis of text across a wide array of subjects. In this module's case however, the range of categories is even more diverse, detailed and nuanced, with over 200 predefined categories. This again required pinpointing specific categories that were relevant to the idea of 'sensationalism' to reduce noise being inserted into the networks architecture. These were:

- **Empath_hate**: Captures expression of hate, which could be sensational or provocative.
- **Empath_aggression**: Reflects aggressive behaviour or language, common in sensationalist narratives.
- **Empath_anger**: Similar to LIWC's anger dimension but broader, capturing more contextual forms of anger.
- **Empath_neglect**: Indicates content that plays on fears of being overlooked or mistreated, which can be a sensational angle.
- **Empath_suffering**: Reflects language related to suffering, which can be used to evoke strong empathetic responses.
- **Empath_fear**: Measures expressions of fear, a powerful emotional trigger in sensational content.
- **Empath_disgust**: Indicates expressions of disgust, which can be used to evoke strong negative emotions.
- **Empath_negative_emotion**: A broad measure of negative emotional content, likely prevalent in sensationalism to evoke a strong reaction.
- **Empath_crime**: Captures language related to criminal activities, law enforcement and justice used to grab the reader's attention.
- **Empath_dispute**: Reflects language indicative of conflicts and arguments, identifying texts that emphasise discord and contention.
- **Empath_nervousness**: Measures expressions of anxiety, stress and nervousness, identifying texts that report feelings of unease and apprehension.
- **Empath_terrorism**: Includes language pertaining to terrorism, extremist activities and related security concerns, which are often focal points in sensational journalism to evoke fear amongst readers.

There are many crossovers between the two handpicked categories from the LIWC and Empath python analyses. The reasons for this are firstly due to the open-source nature of the Empath module, allowing for the replication of the neural network for alternate datasets outside of the LOCO dataset. In addition to using both techniques to affirm statistical differences between the two groups of mainstream and conspiracy texts.
All of these values are of a continuous nature, and the testing size for these statistical tests being incredibly large, we can assume a normal distribution and perform T-tests. The results are summarised in the table below.

	T-test value	P-value
LIWC-affect	33.52	6.74×10^{-243}
LIWC-posemo	8.63	6.33×10^{-18}
LIWC-negemo	36.31	5.34×10^{-284}
LIWC-anx	1.93	5.31×10^{-2}
LIWC-anger	42.99	0.00×10^{-0}
LIWC-sad	-7.48	7.61×10^{-14}
LIWC-discrep	-1.85	6.45×10^{-2}
LIWC-differ	-0.94	3.49×10^{-1}
Empath-hate	21.68	1.34×10^{-103}
Empath-aggression	24.05	7.02×10^{-127}
Empath-anger	6.7	2.04×10^{-11}
Empath-neglect	8.78	1.72×10^{-18}
Empath-suffering	7.02	2.25×10^{-12}
Empath-fear	4.68	2.90×10^{-6}
Empath-disgust	15.84	2.64×10^{-56}
Empath-negative-emotion	6.35	2.25×10^{-10}
Empath-crime	25.47	5.85×10^{-142}
Empath-dispute	11.97	5.73×10^{-33}
Empath-nervousness	-0.09	9.30×10^{-1}
Empath-terrorism	33.31	5.98×10^{-240}

TABLE 3
T-test results summary for LIWC and Empath features.

This summary of statistic does confirm the fact that conspiracy articles use sensationalism to a greater extent compared to mainstream documents. This is shown by every test with the exception of LIWC_sad and LIWC_discrep having a positive T-test value, highlighting conspiracy scoring much higher in these categories. In particular, LIWC_hate, Empath_crime and Empath_terrorism all received incredibly high T-test value highlighting the huge difference between the two groups, in addition to low P-values casting out any doubt that these differences are due to random chance.
With this final trait confirmed, we can conclude that the traits reported of conspiracy documents do ring true, and can therefore securely embedded into a neural network with the intention of classifying documents as either conspiracy or mainstream.

3.2 Neural network implementation

3.2.1 Merged dataframe

The first step in implementing the neural network proposed by FNDNet is to initially curate the dataset that would be inserted into the system. The initial LOCO dataset has a lot of information and features that are obsolete, for example the countless LIWC and Empath features as well as information

provided such as social media shares, which upon further inspection, showed redundant information as many articles were missing . Doing this was essential to reduce any noise that would overshadow any potential learning that could take place.

Firstly we created a merged dataframe of the key features based on the common document ID that the latent features dataframe, and the actual textual content and titles of each document. Then by using a basic label encoder from SciKit learn, the subcorpus column containing each documents label was encoded into a 0 for conspiracy, and a 1 for mainstream. With this the word embeddings were ready to be calculated for all text and titles. As detailed earlier in section 2 (related work), there are multiple different algorithms that are commonly used to convert textual content into a matrix format. We chose to use Global Vectors (GloVe), the unsupervised learning algorithm for obtaining vector representations for words.

3.2.2 Word embeddings calculation

The specific GloVe algorithm used was Stanford's implementation, which is renowned for its efficiency and effectiveness in generating word embeddings. The version deployed was built to scale across massive datasets, specifically the English Gigaword Fifth edition, by leveraging optimized algorithms and data structures to process global word-word co-occurrence statistics. By employing this implementation within the Gensim framework, we were able to harness the power of Stanford's GloVe to generate rich, multidimensional word vectors. These vectors not only encapsulate the semantic and syntactic nuances of the language found in the corpus but also enable more nuanced and sophisticated natural language understanding and processing tasks. The integration with Gensim, a popular Python library for NLP, further simplifies the workflow for training, saving, and loading these word vector.

The English Gigaword Fifth edition is a comprehensive text corpus designed specifically for NLP research, with this edition including text data from seven different sources including newspapers, news agencies and broadcast news transcripts, making this a suitable corpus for the task at hand. There are over 6 billion words in the corpus, making it adequate for training.

3.2.3 Loading the data into PyTorch tensors

The majority of the data were the Empath, and LIWC features and these were of the float data type. These were straightforward to convert into PyTorch tensors by using the PyTorch frameworks Torch.tensor attribute. The encoded labels were required to be converted to the format torch.long. This is as one of the most common operations involving labels in machine learning tasks, especially in classification, is indexing into some other tensor, like looking up embeddings or selecting specific rows from a matrix. Indexing operations in PyTorch require the indices to be of an integer type. `torch.long` is used to ensure compatibility with indexing operations across various platforms and to avoid potential type mismatch errors.

Dealing with the word embedding vectors required extra consideration as these are not standard numerical values, rather are in the form of matrices and more notably, of varying lengths. The list of floats are initially converted into NumPy arrays, allowing for a wide range of numerical operations that are efficient and convenient for manipulation and transformation of numerical data. In addition, PyTorch can directly convert NumPy arrays into tensors, making this step a bridge between the initial string format the Gensim GloVe algorithm returns the word embeddings to us, and the final tensor format required for PyTorch modules. As each word embedding was calculated individually for each unique title and body of text, their sizes are not consistent. To remedy this, dynamic padding is performed on the matrices with a 'collate_fn' function, used to pad the title and text vectors within a batch so that all sequences in a particular batch have the same length. This ensures that data is correctly prepared for efficient and effective training and inference.

3.2.4 Hybrid LSTM-CNN network

The first two networks created are discussed in length in the related work section. The first is the base FNDNet, and the second being the FNDNet with the additional word embedding vectors incorporated. The unique model created for this project calls for the combining of Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) for enhanced feature extraction. A RNN-CNN is proposed, with the specific RNN used being a Long Short-Term Memory (LSTM) architecture. This is due to it being particularly well-suited for tasks involving sequences of data, making them highly effective for various NLP tasks, especially text classification [44]. Their ability to analyse text data sequence by sequence allows them to capture the context and semantic relationships within text, combined with the features already embedded into the network such as the word embeddings and Empath features. The equation below details how LSTMs process information, enabling them to make complex decisions based on both the current input and the history of received outputs.

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\ o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t * \tanh(C_t) \end{aligned} \tag{7}$$

To evaluate this model it was first be deployed and trained on the Kaggle dataset to initially replicate the results reported by [14]. Then it was trained on the LOCO dataset with multiple variations of FNDNet + LIWC / EMPATH, plus max pool, and finally the CNN-LSTM model shown in fig 10.

4 Results

4.1 Reproducing FND results

The first call of action before deploying the neural network onto the LOCO dataset, was to recreate the base network and recreate the 98.6% classification rate that was achieved by the authors of the original paper. The authors took part

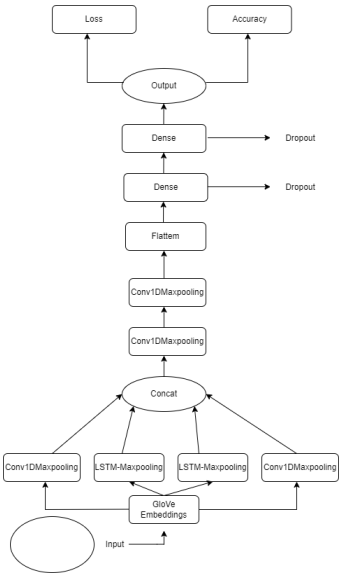

Fig. 10. Computational graph for the CNN-LSTM hybrid model.

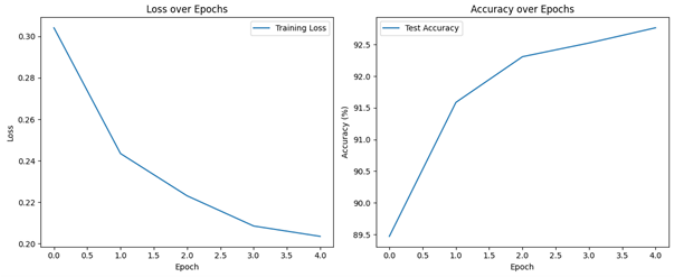

Fig. 11. Results showing reproduced results from the original Fake News Detector papers network on the Kaggle dataset.

in the Kaggle challenge and so had a test and train set provided to them. The testing set provided did not contain labels to ensure those who enter the competition produce labels for the test set and then have them be evaluated by the competition organisers. Because of this, we performed a test-train split on just the train-set that was provided, of which consisted of 20,000 articles. Despite this, when recreating the FND network in PyTorch an accuracy of 93.8% was still achieved after running 5 epochs, showing the strength of this network.

4.2 Affirmation of conspiracy traits

A large portion of this project was firstly seeing if there are definitive differences between conspiracy and mainstream documents. The initial paper that reviewed the LOCO dataset did list out differences, then linguistic and social science professionals compared the two groups, however this was on a small scale. The primary technique used to compare the two groups was Krippendorff's alpha, developed by Klaus Krippendorff. This is a reliability coefficient that measures the agreement among raters who assess a set of items in terms of how well raters agree beyond chance. On top of this, Fleiss' kappa was also used, a statistical measure to assess the reliability of agreement between a fixed number of raters when classifying subjects or items into categories.

Building upon these experiments was imperative to definitively conclude the differences between these two categories of text, especially as these tests were done on small subcorpuses of 40 documents from each sub-group. The methodology section lays out conclusive and solid evidence of the differences between conspiracy and mainstream documents. It can be confidently stated that these two groups of documents have substantial differences that mean that a neural network could accurately classify these documents.

4.3 Neural network models results

When moving to work with the LOCO dataset, some preprocessing was needed before converting the dataset into PyTorch tensors. Firstly dwindling down the 300+ LIWC and Empath features down to the relevant numerical features. The features chosen correlated to articles syntactic and semantic understanding of which relate to the research performed by [4]. The word embeddings for every title and article were then calculated to be able to replicate the NLP techniques that were utilised in the original FND network architecture as proposed by [14]. Then the function *'pad_sequence'* from the *'torch.nn.utils.rnn'* submodule in PyTorch was used to pad a sequence of variable-length tensors with zeros to make them all the same length. Finally a 20/80 test train split was performed to allow a substantial amount of data for each model to be able to learn from, whilst leaving plenty of articles to be able to test the performance.

To evaluate the success of each model, various metrics were used:

- **Accuracy**: Ratio of correctly predicted articles, both true positives and true negatives.
- **Precision**: Ratio of correctly predicted positive articles to the total predicted positives.
- **Recall**: Ratio of correctly predicted positive articles to all observations in the actual class.
- **f1_score**: The weighted average of precision and recall, so this score takes both false positives and false negatives into account.
- **Specificity**: Ratio of correctly predicted negative articles to all articles in the actual negative class.

In the context of this project, specificity is the most important score calculated, as we are primarily interested in indentifying articles which are conspiracy and so they can be fact-checked and looked over before being released to the wider public.

4.3.1 FNDNet with LIWC and Empath features

The initial model tested incorporated both LIWC and Empath NLP features to evaluate the model's performance. The loss curve indicated progressive learning, as evidenced by the decreasing loss values with each epoch. Including LIWC features serves to establish a baseline for the performance of the NLP techniques utilized. This baseline is critical for comparing scenarios where LIWC features might be excluded, especially since using LIWC requires a license for analysis on external datasets.

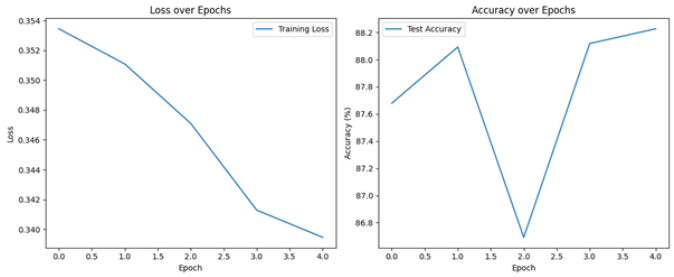

Fig. 12. Results showing reproduced results of the FNDNet with LIWC and Empath features on the LOCO dataset.

TABLE 4
Classification Performance Metrics

Metric	Score
Accuracy	*88.2%*
Precision (Mainstream)	*90.2%*
Recall (Mainstream)	*94.6%*
F1 Score (Mainstream)	*92.3%*
Specificity (Conspiracy)	*68.7%*

4.3.2 *FNDNet with just Empath*

Here this model is identical to the one prior with the one key difference. The LIWC features are omitted to establish how the Empath features performed alone. As shown in

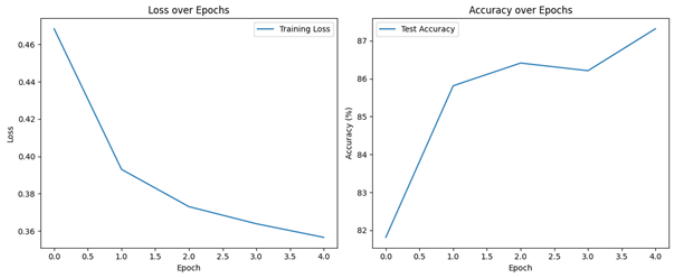

Fig. 13. Results showing reproduced results of the FNDNet with just Empath features on the LOCO dataset.

TABLE 5
Classification Performance Metrics

Metric	Score
Accuracy	*87.3%*
Precision (Mainstream)	*88.2%*
Recall (Mainstream)	*96.0%*
F1 Score (Mainstream)	*92.0%*
Specificity (Conspiracy)	*60.7%*

the table, this only resulted in a minor drop in accuracy of just 0.9% point overall, showing that LIWC analysis is not essential. However, a significant drop is noticed in the specificity which is the most significant metric. This does show the significantly higher strength of the LIWC analysis compared to the Empath NLP tool.

4.3.3 *FNDNet with Maxpool*

The Maxpool layer in PyTorch is a layer utilised in CNNs to downsample or reduce the dimensionality of the input it receives. Implementing such a layer did again decrease the

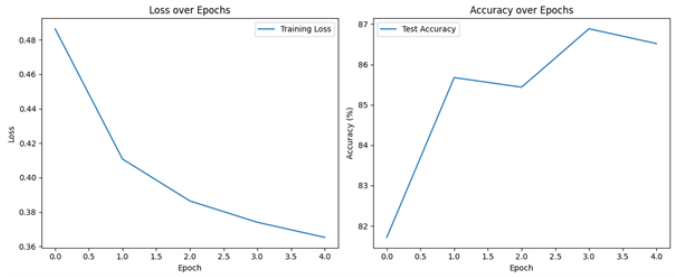

Fig. 14. Results showing reproduced results of the FNDNet with just Empath features and Maxpool layers on the LOCO dataset.

TABLE 6
Classification Performance Metrics

Metric	Score
Accuracy	*86.5%*
Precision (Mainstream)	*86.0%*
Recall (Mainstream)	*98.2%*
F1 Score (Mainstream)	*91.7%*
Specificity (Conspiracy)	*50.8%*

overall accuracy by 0.8% with a huge loss of performance seen in the specificity, showing conspiracy articles were correctly classified at a much lower rate.

4.3.4 *CNN-LSTM model results with MaxPool and Empath features*

The final model to run was the hybrid model, combining the strengths of the CNN with an RNN, specifically the LSTM. The results here show very similar results to what

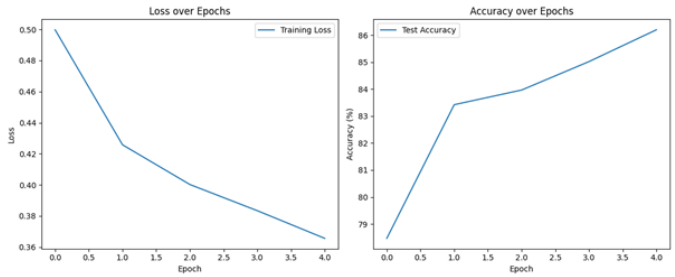

Fig. 15. Results showing reproduced results of the CNN-LSTM model results with MaxPool and Empath features on the LOCO dataset.

TABLE 7
Classification Performance Metrics

Metric	Score
Accuracy	*86.2%*
Precision (Mainstream)	*90.0%*
Recall (Mainstream)	*92.0%*
F1 Score (Mainstream)	*91.0%*
Specificity (Conspiracy)	*68.7%*

was seen in previous models, however a significant increase in the results for the specificity. In fact it holds the highest score for the specificity value since the initial model that was deployed.

4.3.5 CNN-LSTM model with reduced dropout (0.3)

The final model tested was a replica of the hybrid model, only with a reduction in the dropout score between each of the linear layers. This was to preserve information and learning between the neurons of each layer. The overall

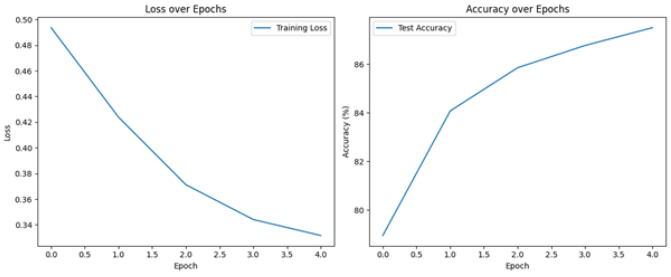

Fig. 16. Results showing reproduced results of the CNN-LSTM model results with MaxPool and Empath features with reduced dropout on the LOCO dataset.

TABLE 8
Classification Performance Metrics

Metric	Score
Accuracy	*87.5%*
Precision (Mainstream)	*88.2%*
Recall (Mainstream)	*96.3%*
F1 Score (Mainstream)	*92.0%*
Specificity (Conspiracy)	*60.5%*

accuracy slightly increases, showing that the model overall tends to perform better, insinuating the regularisation techniques such as dropout can be utilised to improve model performance. However once again, reducing complexity affected the model's ability to accurately classify articles labelled conspiracy, which is the priority for this model.

5 Evaluation

5.1 Traits of conspiracy

When evaluating this project, it is essential to recognize the two significant components that formed the basis of its structure. **First** was determining whether an objective difference existed between the two articles labelled conspiracy and mainstream. This was essential as these articles were collated by Miani [25] using websites and assigning them ground labels rather than manually checking each document. This is what allowed such a substantial dataset to exist of 88 million words. Then the external review done by Mompelat [4], as previously mentioned was performed on incredibly small sub-samples of just 80 articles in total. This is compared to the 96203 articles that were present in the dataset.

With this understanding, evaluating the several key features as presented in table 1 of conspiracy articles found by this external review was necessary to be able to hand over the task of classifying these documents to a machine which lacks the contextual, emotional and critical understanding required to understand text. As presented in the methodology section, all features were tested on a sub-sample of 40,000 articles, with 20,000 from each article type. The results from this came out as anticipated, confirming the research done by [4], with several statistical tests done. This showed a statistically significant difference between the two groups showing there was a very small chance these differences were due to randomness.

Conspiracy articles are confirmed to have the following features:

- Ask more questions - up to 3x as many on average.
- Capitalise words more often than mainstream. (Conspiracy mean = 0.0190, mainstream mean = 0.0157)
- Tend to follow unconventional rules of punctuation as laid out by open-source proof-reading software. Conspiracy articles tend to break these twice as often as mainstream.
- Tend to speak in a more personal manner, using first and second person pronouns up to twice as often.
- Mention conspiracy theories in the article itself up to 50% more often than mainstream articles.
- Use sensationalism to draw readers into the contents of the article with themes of terrorism, fear and anger being more statistically significant.

5.2 Neural network implementation

With all these traits confirmed and an objective measurable difference discovered, the **second** part of this project was to move onto creating neural network models that would be able to classify articles as either conspiracy or mainstream. Results in this case were not as desirable as initially hoped, however overall the significant work presented in this project does show that neural networks do in fact have the ability to direct fact-checkers to articles that may contain misinformation.

The work done by Kaliyar [14] which formed the basis of the neural network architecture produced was replicated and similar results to the original report. Due to the pitfall of the original test set the authors used to evaluate their model being restricted by the Kaggle competition in which the paper was submitted, an exact replica of their work could not be performed. However, performing an additional test-train split on the training set that was publicly available did allow for results of 93.8% accuracy to be achieved when running the model on the Kaggle Fake News Dataset. This shows the validity and usability of the proposed architecture.

When looking at the breakdown of the results, the model consistently performs better at predicting mainstream articles as opposed to conspiracy documents. This could be due to a lower sample size of conspiracy documents. Alternatively it could be due to the inherent traits of conspiracy articles. While mainstream documents have proven to be more likely to follow the conventional rules of the English language, the tendency for conspiracy documents to abandon these patterns may make locating trends and patterns in such documents more difficult due to their ambiguous nature.

The results of the hybrid model did outperform the others as expected, where the best performing model was the CNN-LSTM with a reduced dropout rate, achieving an accuracy 87.5%. Possibly with more epochs all the models could perform even better as the loss curve was shown to consistently decrease, though plateauing around the 5 epoch mark, meaning the scope of increase for the performance would

naturally be limited. Where the hybrid model excelled was the specificity score, relating to the classifying of conspiracy articles, where the highest score of 68.7% was achieved. This could be due to the enhanced feature extraction provided by the LSTM networks placed in the beginning of the network before concatenation occurs.

5.3 Limitations

The constraints faced during the initial stages of this project, encompassing both time limitations and unforeseen personal circumstances, inevitably led to a focus on the implementation and evaluation of a narrower set of models than originally planned. While the initial portion of the project was successfully completed, providing valuable insights into the differences between both types of articles, the full scope of exploratory work regarding neural network implementation remains untapped.

This unexplored potential offers a rich avenue for future research. Given the preliminary success of the models tested, additional work could involve experimenting with a wider array of neural network architectures and configurations. For instance, the usage of other hybrid models combining different neural network types could yield further improvements in classification accuracy and robustness.

Further experimentation could also explore the impact of different preprocessing techniques, feature extraction methods, and data augmentation strategies on model performance. These experiments would not only refine the effectiveness of the models but also contribute to a deeper understanding of the dynamics between model architecture and data characteristics.

6 Conclusion

This project has successfully created neural networks to classify documents and confirmed the existence of a statistically significant difference between conspiracy and mainstream documents. This verification is imperative for the future research into this field of work as it shows that Artificial Intelligence techniques like deep learning can be used to identify articles that could contain misinformation with the intent to cause harm.

This data analysis was performed over an incredibly large sample size of 40,000 documents, which not only strengthens the statistical power of the analysis but also ensures a comprehensive representation of diverse document types and sources. This breadth and depth of data reduce the influence of chance and help mitigate biases that the smaller subset of the initial review faced.

These findings not only highlight the efficacy of deep learning models in distinguishing between conspiracy and mainstream documents, but also suggest a potential pathway for leveraging these technologies in real-world applications. For instance, social media platforms and news organizations could integrate these models into their content moderation systems to more effectively flag and review potentially harmful misinformation before it spreads.

Furthermore, the ability to accurately classify such documents opens the door for more nuanced research into the characteristics that differentiate conspiracy theories from factual reporting. Future studies could explore the linguistic, thematic, and emotional cues that these models identify as indicators of misinformation. This could lead to a better understanding of how conspiracy theories propagate and why they are appealing to certain audiences.

Continued experimentation and refinement of the models is essential to enhance their effectiveness and accuracy. Specifically, the CNN-LSTM architecture, which combines the spatial feature extraction capabilities of Convolutional Neural Networks (CNNs) with the sequence processing strength of Long Short-Term Memory (LSTM) networks, shows promise for document classification tasks. However, the initial findings indicated that including Maxpool layers may detrimentally impact the accuracy of conspiracy document classification. This suggests that the pooling layers might be overly simplifying the feature maps generated by the CNN, thereby losing critical information that is useful for distinguishing subtle nuances in conspiracy texts.

To address this, future work could involve removing the Maxpool layers to allow the model to maintain more of the detailed features extracted from the documents. This modification could potentially enhance the model's sensitivity to the complex patterns often found in conspiracy-related content, which may be smoothed over by the downsampling effect of Maxpooling.

Moreover, increasing the number of training epochs could further improve the model's performance. More epochs would allow the CNN-LSTM model more iterations to learn from the data, adjusting its weights and biases to better capture the distinctions between conspiracy and mainstream documents. It is essential, however, to monitor for overfitting as the number of epochs increases. Implementing techniques such as early stopping, where training can be halted when validation accuracy ceases to improve, can help mitigate this risk.

Overall this project was a success and has laid out the groundwork for future studies and work in tackling the spread of misinformation, being one of the greatest threats we face in the 21st century.

REFERENCES

[1] Zoë Adams, Magda Osman, Christos Bechlivanidis, and Björn Meder. (why) is misinformation a problem? *Perspectives on Psychological Science*, page 17456916221141344, 2023.

[2] Luigi De Angelis, Francesco Baglivo, Guglielmo Arzilli, Gaetano Pierpaolo Privitera, Paolo Ferragina, Alberto Eugenio Tozzi, and Caterina Rizzo. Chatgpt and the rise of large language models: the new ai-driven infodemic threat in public health. *Frontiers in Public Health*, 11:1166120, 2023.

[3] M Giulia Napolitano and Kevin Reuter. What is a conspiracy theory? *Erkenntnis*, 88(5):2035–2062, 2023.

[4] Ludovic Mompelat, Zuoyu Tian, Amanda Kessler, Matthew Luettgen, Aaryana Rajanala, Sandra Kübler, and Michelle Seelig. How "loco" is the loco corpus? annotating the language of conspiracy theories. In *Proceedings of the 16th Lingusitic Annotation Workshop (LAW-XVI) within LREC2022*, pages 111–119, 2022.

[5] Y Linlin Huang, Kate Starbird, Mania Orand, Stephanie A Stanek, and Heather T Pedersen. Connected through crisis: Emotional proximity and the spread of misinformation online. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*, pages 969–980, 2015.

[6] Howard Rosenberg and Charles S Feldman. *No time to think: The menace of media speed and the 24-hour news cycle*. A&C Black, 2008.

[7] HK Khalifa, Sherif A Badran, MS Al-Absy, and QA Almaamari. Social media and spreading the news of covid-19 pandemic in the arab world. *International Journal on Emerging Technologies*, 11(5):680–685, 2020.

[8] Edson C Tandoc Jr. The facts of fake news: A research review. *Sociology Compass*, 13(9):e12724, 2019.

[9] Kai H Lim, Izak Benbasat, and Lawrence M Ward. The role of multimedia in changing first impression bias. *Information Systems Research*, 11(2):115–136, 2000.

[10] Briony Swire-Thompson, Joseph DeGutis, and David Lazer. Searching for the backfire effect: Measurement and design considerations. *Journal of applied research in memory and cognition*, 9(3):286–299, 2020.

[11] Fabio Giglietto, Laura Iannelli, Augusto Valeriani, and Luca Rossi. 'fake news' is the invention of a liar: How false information circulates within the hybrid news system. *Current sociology*, 67(4):625–642, 2019.

[12] Ryan L Boyd, Ashwini Ashokkumar, Sarah Seraj, and James W Pennebaker. The development and psychometric properties of liwc-22. *Austin, TX: University of Texas at Austin*, pages 1–47, 2022.

[13] Ethan Fast, Binbin Chen, and Michael S Bernstein. Empath: Understanding topic signals in large-scale text. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 4647–4657, 2016.

[14] Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. Fndnet–a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61:32–44, 2020.

[15] Jamal Abdul Nasir, Osama Subhani Khan, and Iraklis Varlamis. Fake news detection: A hybrid cnn-rnn based deep learning approach. *International Journal of Information Management Data Insights*, 1(1):100007, 2021.

[16] Muhammad F Mridha, Ashfia Jannat Keya, Md Abdul Hamid, Muhammad Mostafa Monowar, and Md Saifur Rahman. A comprehensive review on fake news detection with deep learning. *IEEE Access*, 9:156151–156170, 2021.

[17] You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. Toward computational fact-checking. *Proceedings of the VLDB Endowment*, 7(7):589–600, 2014.

[18] Kai Shu, H Russell Bernard, and Huan Liu. Studying fake news via network analysis: detection and mitigation. *Emerging research challenges and opportunities in computational social network analysis and mining*, pages 43–65, 2019.

[19] Cannannore Nidhi Kamath, Syed Saqib Bukhari, and Andreas Dengel. Comparative study between traditional machine learning and deep learning approaches for text classification. In *Proceedings of the ACM Symposium on Document Engineering 2018*, pages 1–11, 2018.

[20] Chenguang Song, Nianwen Ning, Yunlei Zhang, and Bin Wu. A multimodal fake news detection model based on crossmodal attention residual and multichannel convolutional neural networks. *Information Processing & Management*, 58(1):102437, 2021.

[21] Dhruv Khattar, Jaipal Singh Goud, Manish Gupta, and Vasudeva Varma. Mvae: Multimodal variational autoencoder for fake news detection. In *The world wide web conference*, pages 2915–2921, 2019.

[22] Shashank Gupta, Raghuveer Thirukovalluru, Manjira Sinha, and Sandya Mannarswamy. Cimtdetect: A community infused matrix-tensor coupled factorization based method for fake news detection. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 278–281, 2018.

[23] Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu. Unsupervised fake news detection on social media: A generative approach. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5644–5651, 2019.

[24] Qi Su, Mingyu Wan, Xiaoqian Liu, Chu-Ren Huang, et al. Motivations, methods and metrics of misinformation detection: an nlp perspective. *Natural Language Processing Research*, 1(1-2):1–13, 2020.

[25] Alessandro Miani, Thomas Hills, and Adrian Bangerter. Loco: The 88-million-word language of conspiracy corpus. *Behavior research methods*, pages 1–24, 2021.

[26] Karen M Douglas and Robbie M Sutton. What are conspiracy theories? a definitional approach to their correlates, consequences, and communication. *Annual review of psychology*, 74:271–298, 2023.

[27] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[28] Tanja Pavleska, Andrej Školkay, Bissera Zankova, Nelson Ribeiro, and Anja Bechmann. Performance analysis of fact-checking organizations and initiatives in europe: a critical overview of online platforms fighting fake news. *Social media and convergence*, 29:1–28, 2018.

[29] Clayton Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, volume 8, pages 216–225, 2014.

[30] Kenneth Ward Church. Word2vec. *Natural Language Engineering*, 23(1):155–162, 2017.

[31] Abhishek Mallik and Sanjay Kumar. Word2vec and lstm based deep learning technique for context-free fake news detection. *Multimedia Tools and Applications*, 83(1):919–940, 2024.

[32] Yann LeCun, Koray Kavukcuoglu, and Clément Farabet. Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE international symposium on circuits and systems*, pages 253–256. IEEE, 2010.

[33] Long Guo, Dongxiang Zhang, Lei Wang, Han Wang, and Bin Cui. Cran: a hybrid cnn-rnn attention-based model for text classification. In *Conceptual Modeling: 37th International Conference, ER 2018, Xi'an, China, October 22–25, 2018, Proceedings 37*, pages 571–585. Springer, 2018.

[34] Jin Zheng and Limin Zheng. A hybrid bidirectional recurrent convolutional neural network attention-based model for text classification. *IEEE Access*, 7:106673–106685, 2019.

[35] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detection of online fake news using n-gram analysis and machine learning techniques. In *Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments: First International Conference, ISDDC 2017, Vancouver, BC, Canada, October 26-28, 2017, Proceedings 1*, pages 127–138. Springer, 2017.

[36] Nicole O'Brien, Sophia Latessa, Georgios Evangelopoulos, and Xavier Boix. The language of fake news: Opening the black-box of deep learning based detectors. 2018.

[37] Bilal Ghanem, Simone Paolo Ponzetto, Paolo Rosso, and Francisco Rangel. Fakeflow: Fake news detection by modeling the flow of affective information. *arXiv preprint arXiv:2101.09810*, 2021.

[38] Vivek Singh, Rupanjal Dasgupta, Darshan Sonagra, Karthik Raman, and Isha Ghosh. Automated fake news detection using linguistic analysis and machine learning. In *International conference on social computing, behavioral-cultural modeling, & prediction and behavior representation in modeling and simulation (SBP-BRiMS)*, pages 1–3, 2017.

[39] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[40] Douglas M Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[41] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[42] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.
[43] Sang Gyu Kwak and Jong Hae Kim. Central limit theorem: the cornerstone of modern statistics. *Korean journal of anesthesiology*, 70(2):144, 2017.
[44] Chunting Zhou, Chonglin Sun, Zhiyuan Liu, and Francis Lau. A c-lstm neural network for text classification. *arXiv preprint arXiv:1511.08630*, 2015.