西南科技大学
Southwest University of Science and Technology

本科毕业设计（论文）

题目名称：Machine Learning Based Fake News Detection.

学 院 名 称	School of Computer Science and Technology
专 业 名 称	Computer Science and Technology
学 生 姓 名	MD MAHMUDUL HASAN TAREQ
学 号	4420190083
指 导 教 师	Dr. Yin Long

二〇二三年六月

本科毕业设计（论文）学术诚信声明

本人郑重声明：所呈交的毕业设计（论文），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

Undergraduate Graduation Project (Thesis) Statement of Academic Integrity

I solemnly declare that the graduation project (thesis) submitted is the result of my independent research work under the guidance of my supervisor. Except for those already cited in the article, this paper does not contain the results of any other work that has been published or written by any other individual or collective. Individuals and collectives who have made significant contributions to the study of this article are clearly identified in the text. I am fully aware that the legal consequences of this statement are my responsibility.

作者签名 Author's signature：

日期：　　年　月　日

本科毕业设计（论文）版权使用授权书

本毕业设计（论文）作者同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权西南科技大学可以将本毕业设计（论文）的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本毕业设计（论文）。

Undergraduate graduation project (thesis) copyright authorization letter

The author of this graduation project (thesis) agrees that the school shall retain and send copies and electronic versions of the thesis to the relevant state departments or institutions, allowing the thesis to be consulted and borrowed. I authorize the Southwest University of Science and Technology to compile all or part of the contents of this graduation design (thesis) into the relevant database for retrieval and may use photocopying, micro printing, or scanning to save and compile this graduation design (thesis).

作者签名 Author's signature：　　　　　　　指导教师签名 Instructor's signature：

日期：　年　月　日　　　　　　　　　　　日期：　年　月　日

摘要

在当今的数字时代，假新闻的传播已成为政府、媒体机构和公众关注的主要问题。社交媒体平台的兴起导致虚假和误导性信息的迅速传播，可能对个人和整个社会造成严重后果。为了解决这个问题，机器学习技术已经成为检测假新闻的一种很有前途的方法。

这项研究的主要目标是使用有监督的机器学习算法检测假新闻文章，并找到一个高精度的模型。研究方法包括三个关键步骤：文本预处理、特征提取和假新闻检测。使用 TF-IDF 向量化技术执行特征提取。该模型的训练和评估是在从 Kaggle 收集的综合数据集（即假新闻和真实新闻数据集）上进行的。五种不同的监督机器学习算法，包括决策树分类器、逻辑回归、被动攻击分类器、梯度提升分类器和随机森林分类器，用于模型训练和测试目的。在这些算法中，决策树分类器表现突出，准确率高达 99.57%。使用精度、召回率、F1 分数、准确性、假阳性、真阴性和其他性能指标进一步验证模型性能的评估。这些发现表明，在识别虚假新闻方面有了实质性的改进，超过了之前的最新成果，从而证实了所提出的机器学习技术在检测虚假新闻方面的有效性。

本文还全面回顾了有关假新闻检测的现有文献，包括当前方法的挑战和局限性。研究结果表明，基于机器学习的方法可以在检测假新闻方面实现高精度，但特征和参数的选择会显着影响模型性能。对于未来的工作，也许我们可以提出深度学习模型或具有大型数据集的混合模型。此外，我们可以使用视频或图像数据集来检测假新闻，因为我们都知道这些媒体也用于宣传假新闻。总的来说，这篇论文有助于开发更准确、更可靠的假新闻检测方法，这有助于保护信息的完整性，促进一个更知情、更负责任的社会。

关键词：假新闻检测，TF-IDF 向量化，特征提取，监督学习，评估指标，数据集偏差，实时检测。

ABSTRACT

In today's digital age, the spread of fake news has become a major concern for governments, media outlets, and the general public. The rise of social media platforms has led to the rapid dissemination of false and misleading information, which can have serious consequences for individuals and society as a whole. To address this issue, machine learning techniques have emerged as a promising approach for detecting fake news.

The main objective of this research is to detect fake news articles with supervised machine learning algorithms and to find a model with high accuracy. The research methodology consists of three key steps: text pre-processing, feature extraction, and fake news detection. Feature extraction is performed using the TF-IDF vectorization technique. The model's training and evaluation are conducted on a comprehensive dataset collected from Kaggle namely fake and real news dataset. Five distinct supervised machine learning algorithms, including Decision Tree Classifier, Logistic Regression, Passive Aggressive Classifier, Gradient Boosting Classifier, and Random Forest Classifier, are used for model training and testing purposes. Among these algorithms, the Decision tree classifier exhibits outstanding performance, achieving an impressive accuracy rate of 99.57%. The evaluation of the model's performance is further validated using precision, recall, F1 score, accuracy, false positive, true negative, and other performance metrics. These findings demonstrate substantial improvements in the identification of false news, surpassing previous state-of-the-art results, thus affirming the effectiveness of the proposed machine learning technique for detecting fake news.

The thesis also presents a comprehensive review of the existing literature on fake news detection, including the challenges and limitations of current approaches. The research findings suggest that machine learning-based approaches can achieve high accuracy in detecting fake news, but the choice of features and parameters can significantly affect model performance. For future work maybe we can propose a deep learning model or a hybrid model with a large dataset. Also, we can use a video or image dataset for detecting fake news as we all know these medium also used for promoting fake news. Overall, this thesis contributes to the development of more accurate and reliable methods for detecting fake news, which can help to preserve the integrity of information and promote a more informed and responsible society.

Key word: Fake News Detection, TF-IDF Vectorization, Feature Extraction, Supervised Learning, Evaluation Metrics, Dataset Bias, Real-Time Detection.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Research Background

In the digital age, the amount of information available to us is unprecedented. With the internet at our fingertips, we can access news, opinions, and perspectives from all corners of the globe, and we can do so instantaneously. However, with this vast ocean of information comes a great deal of noise. It can be difficult to discern what is fact from what is fiction, and what is legitimate news from what is fake. This is especially true in the realm of social media, where anyone can post anything, regardless of its veracity. People now consume news more frequently from social media than from traditional news media since social media has become indispensable. According to a survey, 67% of American people in 2017 primarily used social media to consume news [1].

You may have heard that Nancy Pelosi used Social Security funds to pay for the impeachment investigation and that President Trump's father was a KKK member. According to a survey by the nonprofit organization Avaaz, political fake news received more than 150 million views in 2019. As absurd as those claims may appear, they were among the most shared fake news articles on Facebook in 2019. Furthermore, a study conducted by Soroush Vosoughi, PhD, a computer scientist at Dartmouth University, and his coworkers discovered that false information spreads more swiftly and reaches more individuals than the real thing [2]. As for the report by Avaaz and the study by Dartmouth University, they highlight the concerning impact that fake news can have on society. The spread of false information can lead to confusion, mistrust of legitimate sources, and even harm to individuals and communities. Fake news and propaganda have become pervasive, and they can have a profound impact on public opinion and discourse.

Fake news has been around for a long time now, it has existed before the invention of social media platforms. Fake news refers to misinformation, disinformation, or hoaxes spread through traditional or social media. It is a growing concern in today's society, as it can have serious consequences such as influencing public opinion, shaping political outcomes, and even inciting violence. Fake news detection is the process of identifying and verifying the accuracy of news articles and other content. With the explosion of social media and the internet, fake news has become more prevalent and sophisticated, making it increasingly challenging to distinguish between real and fake news.

Fake news detection is a complex and evolving field, and researchers are constantly exploring new techniques and approaches to improve performance and accuracy. While there is no single solution that can detect all instances of fake news, machine learning-based approaches are proving to be highly effective in combating the spread of misinformation and disinformation. The spread of fake news is not a new phenomenon, but its impact has been magnified in recent years due to the rise of social media and the internet. The term "fake news" gained widespread attention during the 2016 U.S. presidential election, where the use of fake news to manipulate public opinion was widely reported [3]. Similar attempts were made during the 2018 Brazilian elections, when WhatsApp was misused to spread disinformation campaigns, making extensive use of altered photos and memes that contained various political criticisms. According to a recent study, 88% of the most popular pictures posted in the month preceding up to the Brazilian elections were either fraudulent or misleading [4].

Fake news has the potential to seriously harm the economy by quickly causing market panic. Fake news about a company's financial performance or a product's safety can cause investors to make bad investment decisions, leading to significant financial losses. Similarly, fake news about a company's reputation or a product's quality can cause a decline in sales, resulting in financial losses for the company. Additionally, fake news can be used to manipulate financial markets by spreading false information about a company or an industry, leading to fluctuations in stock prices or other financial instruments. This type of manipulation can result in financial gains for those spreading the fake news, while causing significant losses for other investors. The "Pizzagate" conspiracy theory in 2016: This fake news story claimed that a Washington D.C. pizza restaurant was a front for a child sex trafficking ring involving top Democrats, including Hillary Clinton. The story went viral on social media and led to a man opening fire inside the restaurant. The incident caused significant damage to the restaurant's reputation and resulted in a decline in sales [5].

(a) A fake news on twitter

(b) It’s impact on the Dow

Figure 1.1 A fake news on Twitter and its impact on the Dow.

A fake news tweet from a hacked Associated Press Twitter account claimed that there had been explosions at the White House and that President Obama had been injured (shown in Figure1.1-(a)). The tweet caused a brief but significant drop in the stock market, resulting in millions of dollars in losses for investors. The Dow dropped more than 140 points (shown in Figure1.1-(b)) in less than six minutes. According to Reuters, the S&P 500's market cap dropped temporarily by a total of $136.5 billion [6].

Fake news can take many forms, including propaganda, conspiracy theories, and hoaxes, and it can be spread through traditional media outlets, social media, and other online platforms. Its effects can be far-reaching, as it can undermine trust in institutions, spread fear and panic, and even incite violence.

As a result, there is an urgent need for tools that can help us identify and filter out false information.

Automated fake news detection has emerged as an important area of research in the field of natural language processing, as it holds great promise for enabling us to better understand and navigate the complex world of online information. While there are many challenges to be overcome in this area, the potential benefits are substantial, and we can be hopeful that continued progress in this field will ultimately lead to a more informed and empowered society. The rise of fake news has led to a growing demand for tools and techniques to detect. Fortunately, technology has provided us with tools to detect fake news. Machine learning-based approaches have emerged as one of the most promising solutions, which allows algorithms to learn patterns and characteristics of fake news and apply those learnings to new information to determine if it is likely to be fake. This approach has been widely adopted by researchers and companies alike, with many developing sophisticated machine learning models that can detect fake news with high accuracy.

1.1.1 Fake news during COVID-19

The COVID-19 pandemic has not only created a health crisis but also a massive infodemic with misinformation spreading rapidly through social media platforms and other online channels. Fake news has become a serious issue, causing harm to individuals and society as a whole. Misinformation about the virus and its transmission, prevention and treatment can result in panic, fear, and mistrust among the public, hindering efforts to control the spread of the disease.

One of the main ways that people spread fake news during the COVID-19 situation is through social media. Social media platforms such as Facebook, Twitter, and Instagram have become a breeding ground for fake news about the virus, with false information and conspiracy theories spreading rapidly through these channels. Additionally, some individuals intentionally spread false information to create confusion or push their own agenda, including promoting unproven treatments, conspiracy theories, or political propaganda. A study from the Reuters Institute for the Study of Journalism found that false and misleading information about COVID-19 was being shared more frequently on social media than accurate information, with a higher likelihood of engagement and shares [7].

Messaging apps such as WhatsApp have also been a significant source of fake news about COVID-19. These apps have become a preferred platform for spreading false information about the virus, with messages forwarded rapidly among groups and individuals. A study from the International Center for Journalists found that WhatsApp was the most popular social media platform for sharing COVID-19-related false information in Brazil, India, and Nigeria.

Fake news during COVID-19 cause significant problems in our society. One of the main problems was that, it lead us to confusion and fear among the public. False information about the transmission, symptoms, and treatment of the virus cause people to take ineffective or even harmful measures to protect themselves. For example, one widely circulated false claim during the pandemic was that drinking bleach or other disinfectants could cure COVID-19 [8]. This dangerous idea gained traction after former US President Donald Trump suggested during a press conference that injecting disinfectants could potentially cure the virus [9]. Despite immediate backlash and warnings from medical professionals, the idea spread quickly on social media, with some people even trying the dangerous remedy and suffering serious harm as a result.

Misinformation about vaccines has also been a significant issue during the pandemic. Some people have spread false claims that COVID-19 vaccines are dangerous, ineffective, or even contain microchips or other tracking devices. These rumors have led to vaccine hesitancy, making it harder to achieve herd immunity and control the spread of the virus. In some cases, vaccine misinformation has led to violent protests or attacks on healthcare workers and vaccination centers.

Fake news also undermine public health measures designed to control the spread of COVID-19. False information about the effectiveness of masks or social distancing lead people to disregard these measures, which contribute to the virus spreading more easily. Another common type of misinformation during the pandemic was conspiracy theories about the origins of the virus. Some people claimed that the virus was engineered in a laboratory, deliberately released, or even that it didn't exist at all. These baseless theories not only sow confusion and distrust but also undermine efforts to identify the true source of the virus and prevent future pandemics. A study published in the journal Nature found that people who are exposed to fake news about COVID-19 are less likely to comply with public health guidelines and more likely to engage in risky behaviors [10].

Finally, during COVID-19 fake news contribute to stigmatization and discrimination against certain groups. For example, false information about the virus being linked to certain ethnic or religious groups lead to xenophobia and racism [10]. During COVID-19 fake news was a significant problem in our society, with false information and conspiracy theories spreading rapidly through social media and messaging apps. This fake news cause confusion, fear, and mistrust in public health authorities, undermine public health measures, erode public trust, and contribute to stigmatization and discrimination.

1.1.2 Machine learning

Machine learning is a subfield of artificial intelligence (AI) that focuses on developing algorithms and models that can learn from data and make predictions or decisions without being explicitly programmed. Machine learning algorithms can be broadly classified into four categories: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.

- Supervised Learning
  Supervised learning is a type of machine learning where the algorithm is trained on a labeled dataset, which means that the correct output is provided for each input. The goal of supervised learning is to learn a mapping from inputs to outputs, so that the algorithm can make accurate predictions on new, unseen data. Supervised learning can be further divided into two categories: classification and regression. In classification, the goal is to predict a categorical variable or a class label, such as whether an email is spam or not. In regression, the goal is to predict a continuous variable, such as the price of a house based on its features.
  The most common supervised learning algorithms include decision trees, logistic regression, support vector machines (SVMs), and artificial neural networks (ANNs). These algorithms can be trained on various types of data, such as text, images, and numerical data.

- Unsupervised Learning
  Unsupervised learning is a type of machine learning where the algorithm is trained on an unlabeled dataset, which means that no output variable is provided for the inputs. The goal of unsupervised learning is to identify patterns or structure in the data.

  Unsupervised learning can be further divided into two categories: clustering and dimensionality reduction. In clustering, the goal is to group similar data points together based on their features. In dimensionality reduction, the goal is to reduce the number of features in the data while preserving as much information as possible.
  The most common unsupervised learning algorithms include k-means clustering, hierarchical clustering, principal component analysis (PCA), and autoencoders. These algorithms can be used for various applications, such as customer segmentation, anomaly detection, and data compression.

- Semi-Supervised Learning
  Semi-supervised learning is a type of machine learning where the algorithm is trained on a combination of labeled and unlabeled data. The goal of semi-supervised learning is to leverage the unlabeled data to improve the accuracy of the model on the labeled data.

  Semi-supervised learning is particularly useful when labeled data is scarce or expensive to obtain. By using the unlabeled data, the model can generalize better to new, unseen data. The most common semi-supervised learning algorithms include self-training, co-training, and multi-view learning. These algorithms can be used for various applications, such as speech recognition, image classification, and natural language processing.

- Reinforcement Learning
  Reinforcement learning is a type of machine learning where the algorithm learns to make decisions based on feedback from its environment. The goal of reinforcement learning is to maximize a reward signal by taking actions that lead to positive outcomes and avoiding actions that lead to negative outcomes. Reinforcement learning is particularly useful for applications where the optimal action may not be known in advance, such as playing games or controlling robots. The algorithm learns from its own experience by trial and error, and adjusts its behavior over time to maximize the reward signal.

  The most common reinforcement learning algorithms include Q-learning, policy gradient methods, and actor-critic methods. These algorithms can be used for various applications, such as game playing, robotics, and autonomous vehicles.

In conclusion, machine learning is a powerful tool for solving complex problems in various domains, such as healthcare, finance, and transportation. Supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning are the four main categories of machine learning algorithms, each with its own strengths and weaknesses. By understanding the differences between these categories and selecting the appropriate algorithm for a given task, machine learning can be applied to a wide range of real-world problems.

1.2 Motivation of the Research

The motivation for research in machine learning-based fake news detection stems from the growing concern over the spread of misinformation and its impact on society. Fake news has become a significant issue in recent years, with the widespread use of social media and the internet as primary sources of news and information. The consequences of fake news can be severe, leading to a lack of trust in traditional media, political polarization, and even threats to public safety. In some cases, fake news has been used to influence elections, incite violence, or spread harmful propaganda.

Detecting fake news manually is a time-consuming and challenging task, and the volume of information available online makes it virtually impossible to fact-check every piece of content. Therefore, the development of automated tools and systems to assist in the detection of fake news is critical.

Machine learning provides a powerful framework for addressing the challenge of fake news detection. By training models on large datasets of labeled examples, machine learning algorithms can learn to recognize patterns and characteristics that distinguish fake news from genuine articles. These models can then be used to flag potential instances of fake news in real-time, assisting journalists, fact-checkers, and social media platforms in combating the spread of misinformation.

Overall, the motivation for research in machine learning-based fake news detection is to provide a solution to the growing problem of fake news and to enable people to make informed decisions based on accurate information. By developing effective tools for detecting and combating fake news, we can help promote a more informed and democratic society.

1.3 Problem Statement

The proliferation of fake news in online platforms poses a significant challenge to the credibility and reliability of information sources. The lack of effective mechanisms to identify and combat fake news undermines public trust, fosters misinformation, and can have adverse social, political, and economic consequences. Manual fact-checking processes are time-consuming, impractical for the vast amount of online content, and subject to human biases. Thus, there is a pressing need for automated systems that leverage machine learning techniques to accurately detect and flag fake news in real-time.

Existing research on machine learning-based fake news detection has made notable progress but faces several key challenges. First, the constantly evolving nature of fake news requires adaptive models capable of recognizing novel patterns and strategies employed by purveyors of misinformation. Second, the ability to distinguish subtle variations in content, including misleading headlines, distorted information, or out-of-context quotes, remains a significant challenge. Third, the lack of comprehensive labeled datasets that accurately represent the diverse range of fake news sources and techniques limits the generalization and effectiveness of developed models.

Moreover, mitigating the impact of fake news requires not only accurate detection but also effective dissemination of the results to the intended audience. Ensuring that fake news detection systems are accessible, user-friendly, and integrated into popular online platforms is crucial for reaching a wide audience and empowering individuals to make informed decisions.

Therefore, the problem at hand is to develop robust and adaptable machine learning-based fake news detection systems that can accurately identify a wide range of fake news instances while minimizing false positives. Additionally, addressing the lack of diverse and comprehensive labeled datasets and incorporating user-friendly interfaces for disseminating detection results are vital aspects of tackling the fake news problem effectively.

By addressing these challenges, the research aims to contribute to the development of reliable and scalable solutions that help combat the spread of fake news, promote information integrity, and foster a more informed and resilient society.

1.4 Research Question

The overarching research question that guides this study is:

"How can machine learning techniques be effectively utilized to detect and combat fake news in online platforms?"

To address this question, the following sub-questions can be explored:

- What are the key linguistic, semantic, and contextual features that can be extracted from textual content to differentiate between genuine and fake news?
- Which machine learning algorithms and models are most suitable for accurately detecting fake news? How can these models be trained and optimized to achieve high detection performance?
- How can the detection models be adapted and updated to effectively identify emerging patterns and evolving strategies used by purveyors of fake news?
- How can the impact and effectiveness of machine learning-based fake news detection systems be evaluated? What evaluation metrics and benchmarks should be used to assess the performance and reliability of these systems?

By addressing these research questions, the study aims to advance our understanding of the effectiveness of machine learning techniques for fake news detection, provide insights into the challenges and limitations of existing approaches, and propose novel methods and strategies to enhance the accuracy and usability of fake news detection systems.

1.5 Research Objective

To develop and evaluate a machine learning model for the detection of fake news in online news articles, with the goal of improving the accuracy and efficiency of fake news identification. This research objective outlines the specific topic of the study (fake news detection), the methodology (machine learning), and the goal of the research (to improve the accuracy and efficiency of fake news identification). To develop and evaluate a machine learning model for the detection of fake news in online news articles with an accuracy rate of at least 90%, a false positive rate of less than 5%, and a processing time of less than 10 seconds per article. The study will use a dataset of at least 20,000 news articles collected from Kaggle, and the model will be evaluated using cross-validation and a separate test dataset.

1.6 Research Scope

- The study will focus on the detection of fake news in online news articles written in English.
- The study will use machine learning algorithms to classify news articles as real or fake based on various features such as the source of the article, the language used, and the content of the article.
- The study will collect a dataset of at least 20,000 news articles, with an equal number of real and fake news articles.
- The study will evaluate the performance of the machine learning model using cross-validation and a separate test dataset.
- The study will not focus on the causes or effects of fake news, but solely on the development and evaluation of a machine learning-based fake news detection model.
- The study will not cover other forms of media such as social media or video content.
- The study will not address the ethical or legal implications of fake news detection or use.

1.7 Thesis Organization

In Chapter 2, we go over "Literature Review" in depth. Our "Methodology" is explained in Chapter 3. Examples include datasets, data preprocessing, and explaining our machine learning model. Following that, in Chapter 4, we give our "Experiments and Results." Then, in Chapter 5, we highlighted our "Discussion and conclusion" and indicated the future scope of the effort. In this way, we conclude our thesis paper.

CHAPTER 2

LITERATURE REVIEW

Detecting fake news on social media is a relatively new idea that has attracted a lot of interest in the literature due to the threats it poses. A lot of intensive effort has been invested into approaching the difficulties from a machine learning or deep learning perspective during the last few years. Based on algorithmic technique, existing methodologies can be grouped into three major categories: traditional machine learning, deep learning, and advanced language model-based detection.

Fake news detection has received a great deal of interest because of its application values. It implements ideas for fake news detection using methods for rumor detection from texts based on similarities between false news and rumor (Zubiaga et al., 2017). Machine learning, particularly deep learning, is an important tool applied for detecting fake news Using machine learning and deep learning (Wu et al., 2016) extract and choose meaningful attributes from data could improve false news identification. Furthermore, the news' contents and network structure can be used to identify fake news.

This study investigates current methods for detecting false news based on content and context, with a focus on news headlines and test items (Kaliyar et al, 2020). The authors (Srivastava et al., 2014) tested their approach on a range of internet sites using rules for content veracity analysis. In one of their tests, they achieved an overall accuracy of 88.00% by using a practical fake news dataset (FakeNewsNet). The Kaggle fake news dataset has been used in the majority of context-related studies to investigate the problem of false news. One study's authors (Ahmed et al., 2017) used TF-IDF (Term Frequency and Inverse Document Frequency) as a feature extractor to detect false news using a variety of machine learning approaches. Authors (Ahmed et al., 2017) in one of their study they used LR (Linear regression) and LVSM (Linear Support Vector Machine). to classify fake news and achieved an accuracy of 89.00% for LR. They achieved 92% accuracy using LVSM.

In their study, the authors (Yang et al., 2018) used CNN (Convolutional Neural Network) to detect fake news. They applied sensitivity analysis in their technique and achieved an accuracy rate of 92.10%. (O'Brien et al., 2018) used deep learning models to classify false news in their study. In their study, they used a DNN (black-box approach) and obtained an accuracy of 93.50%. (Singh et al., 2017) investigated the usage of multiple machine learning techniques for detecting false news (Linguistic Analysis and Word Count). By Using SVM (support vector machine), they were able to attain an accuracy of 87.00 percent. (Kaliyar et al, 2020) stated that for pre-trained word embedding experiments, they substitute the processing layer's parameters with pre-trained word embedding vectors while preserving the indices and freezing the layer to avoid this from changing throughout the gradient descent approach. They were able to get 98.12% using FNDNet. (Purevdagva et al., 2020) proposed an automated framework for the detection of fake political speech that uses different classification methods for extracting features from political speech statement and its metadata including speech subject, location, speaker's profile, speaker’s credibility, and speech context information is proposed.

(Zhou et al., 2019) developed a theory-driven approach to detect fake news by extracting misinformation and clickbait elements from news article text and employing machine learning models such as SVM, Random Forest, XGBoost, Naive Bayes, and Logistic Regression. (Reddy et al., 2020) organize fake news by applying gathering methodologies on stylometric highlights and word vector highlights of the content of news stories with up to 95.49% accuracy.

(Pérez-Rosas et al., 2017) conducted a comparison of humanly and automatically detected fake news using linguistic features such as n-grams, punctuation, psycholinguistic factors, readability, and syntax-based features. Authors used machine learning methods on 57 features, (Gravanis et al., 2019) used AdaBoost and bagging ensemble machine learning algorithms to detect false news utilizing augmented linguistic characteristics with word embedding. The profiles of users can also be utilized to detect fake news. Demographics, social network structure, and user reactions are all considered in social context approaches. Few researchers have looked into fake news detection using social context. (Shu et al., 2018) looked into the relationship between user profiles and fake news. (Shu et al., 2019) look into the role of network properties in the spread of fake news.

Several research investigate the significance of deep learning models over machine learning models for detecting fake news (Choudhary et al., 2021; Mouratidis et al., 2021). Authors (Wang et al., 2017) demonstrates that a hybrid convolutional neural network outperforms machine learning methods in detecting fake news based on language patterns. (Rashkin et al., 2017) obtained good results with LSTM as well. Machine learning baselines were surpassed by models that focused on language features. The bi-directional Gated Recurrent Unit is proposed by (Zhou et al., 2017). The self-attentive system came in first place in the Clickbait Challenge. For false news classification, hybrid deep learning models that blend convolutional and recurrent neural networks fared slightly better than non-hybrid baseline methods (Nasir et al., 2021).

State-of-the-art pre-trained language models, such as BERT (Bidirectional Encoder Representations from Transformers) and ALBERT (A Lite BERT), have gained significant attention due to their exceptional performance across various natural language processing tasks, including fake news classification. According to (Jwa et al., 2019), BERT increased the F-score on classification of fake news by 0.14 when compared to prior state-of-the-art models. Performance of BERT, ALBERT, and similar models in fake news classification has been impressive, often surpassing traditional machine learning approaches. These models have the potential to enhance the development of robust and reliable systems for detecting and combating fake news in various contexts, including social media platforms and news organizations. (Qazi et al., 2020) outperforms a hybrid CNN technique by 15% in terms of accuracy. Author (Aggarwal et al., 2020) present a paper where they achieved an accuracy of 97.02% by using a fine-tuned BERT model for fake news classification.

Significant work has also been done in classifying huge texts (such as articles) as false or true. (Singhania et al., 2017) used 3HAN, a deep learning strategy that involves a three-level bottom-up hierarchical article representation for words, phrases, and headlines, as well as the construction of a vector for each article. This vector is the input to the Neural Network, which is trained on the idea that an article's headline should only describe whether it is phony or not.

In this chapter we presented the related work that is available regarding the Fake News detection. The challenges posed by the linguistic features of news material compelled us to devise a solution. New computer science technologies, such as word embeddings and deep learning, provide us with the tools we need to effectively address the problem. In the following chapter, we will go over the approach we employed and provide some theoretical insights into the methodologies and algorithms we used to conduct our study.

CHAPTER 3

METHODOLOGY

3.1 Research design

The phenomenon of fake news is advancing at a rapid and increasing rate with the expansion of communication and social media. Fake news detection is an emerging research area which is gaining big interest. However, it faces some challenges due to the limited resources such as datasets, processing and analyzing techniques. In this work, we propose a system for Fake news detection that uses machine learning techniques. We used term frequency inverse document frequency (TF-IDF) of bag of as feature extraction technique and we use five different machine learning model to compare each other in terms of their accuracy and evaluation matrix. In this project we use python programming language to build machine learning model for fake news detection. And we downloaded a Datasets called Fake and real news dataset from Kaggle for model training. (https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset). Here is the methodology and general work procedure diagram.

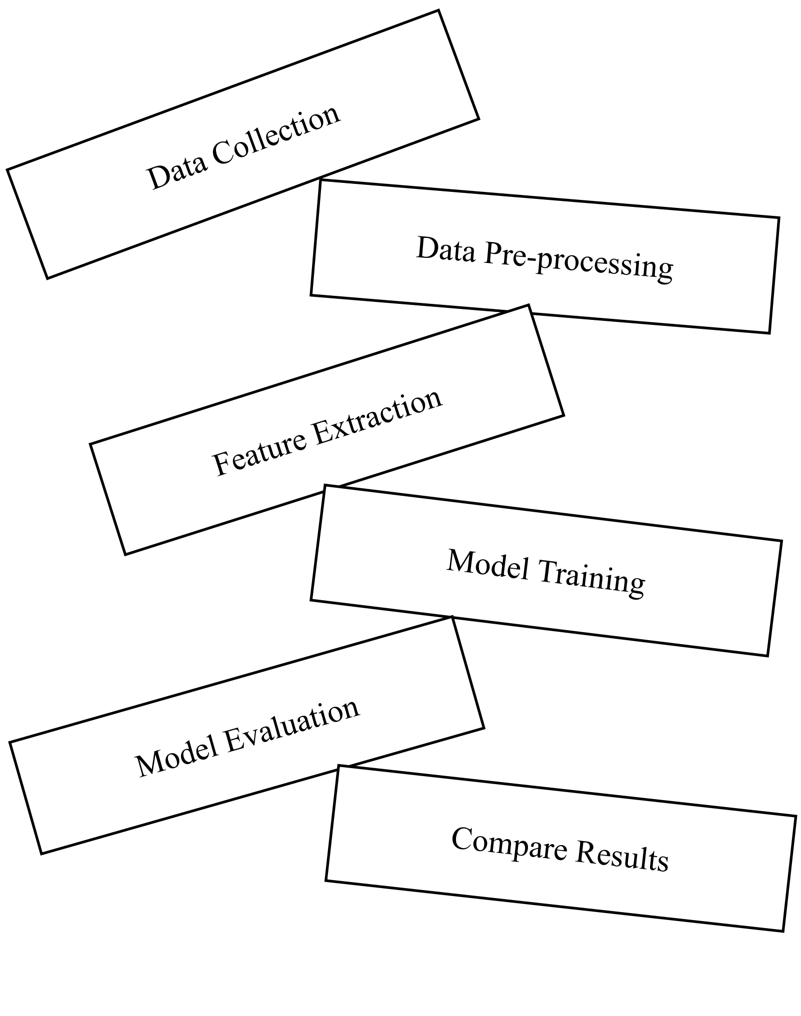

Figure 3.1 Methodology work flow

3.2 Dataset description

The "Fake and Real News Dataset" available on Kaggle was collected and shared by Clément Bisaillon. The dataset consists of two CSV files, one containing "fake news" articles and one containing "real news" articles, collected from a variety of sources. Here is the link of this datasets (https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset) and a brief description of the dataset:

"Fake.csv": This file contains 23,503 articles. The articles were collected from a variety of sources, including hoax websites, satire websites, and misleading news websites.

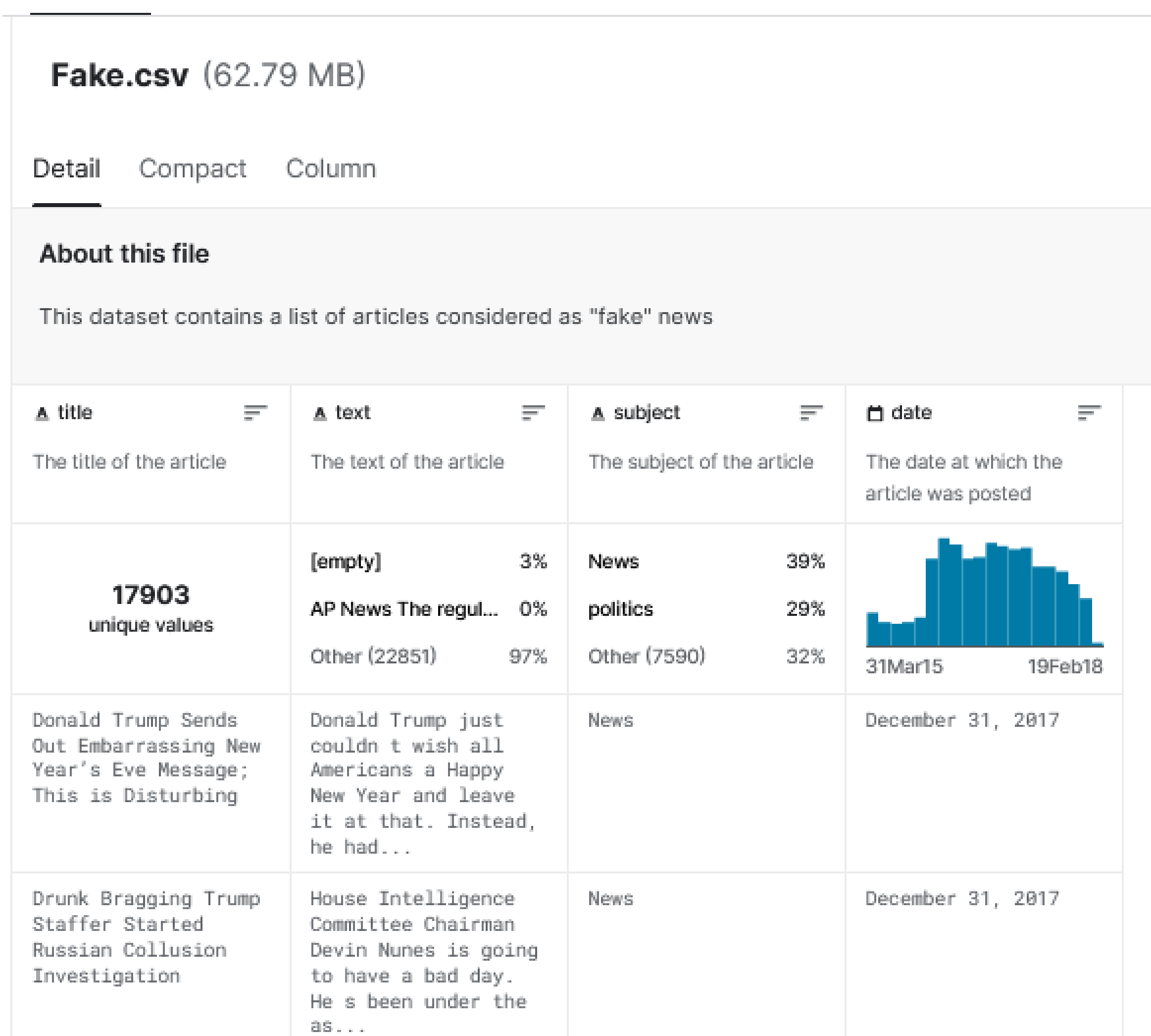

Figure 3.2.1 Fake News Datasets

"True.csv": This file contains 21,418 articles. The articles were collected from a variety of sources, including Reuters, the Associated Press, and other reputable news organizations.

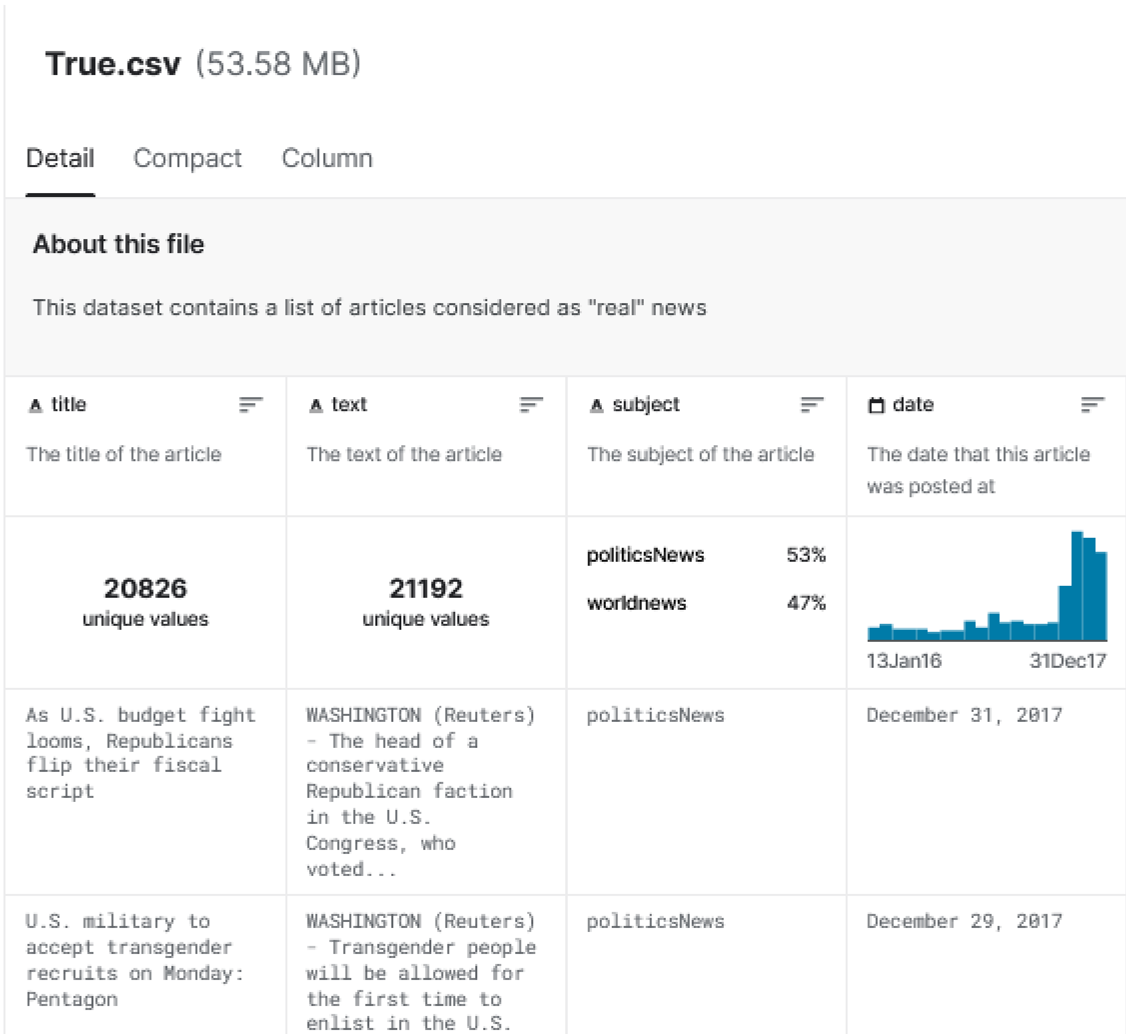

Figure 3.2.2 True News Datasets

Both files contain the following columns:

- "title": The title of the article
- "text": The full text of the article
- "subject": The subject of the article (e.g. politics, worldnews, etc.)
- "date": The date the article was published

The dataset can be used for a variety of purposes, including training and testing machine learning algorithms for fake news detection. However, it's important to note that the dataset is not a comprehensive or representative sample of all fake and real news articles, and there may be biases in the sources that were included. The dataset includes the full text of each article, as well as metadata such as the title, subject, and date of publication. The dataset has been used for several research studies and competitions related to fake news detection. The dataset has some limitations, such as the fact that it only includes articles from a specific time period and that the sources of the "fake news" articles may not be representative of all sources of fake news. However, it is still a useful resource for researchers and practitioners interested in studying fake news detection.

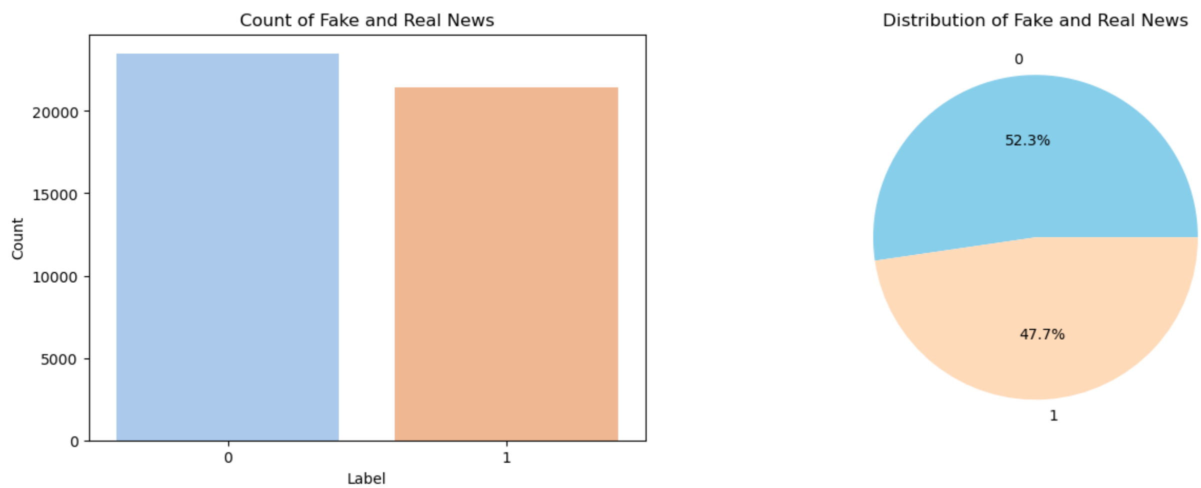

Figure 3.2.3 Count of Fake and Real News and Distribution of Fake and Real News.

In this dataset we have total more than 44,000 articles where 52.3% data are fake news and 47.7% data are real news. Here we have several kinds of news like politics news, world news, news, government news, US news, middle east news etc. Below it the count of different kind of news,

PoliticsNews	11272
worldnews	10145
News	9050
politics	6841
left-news	4459
Government News	1570
US_News	783
Middle-east	778

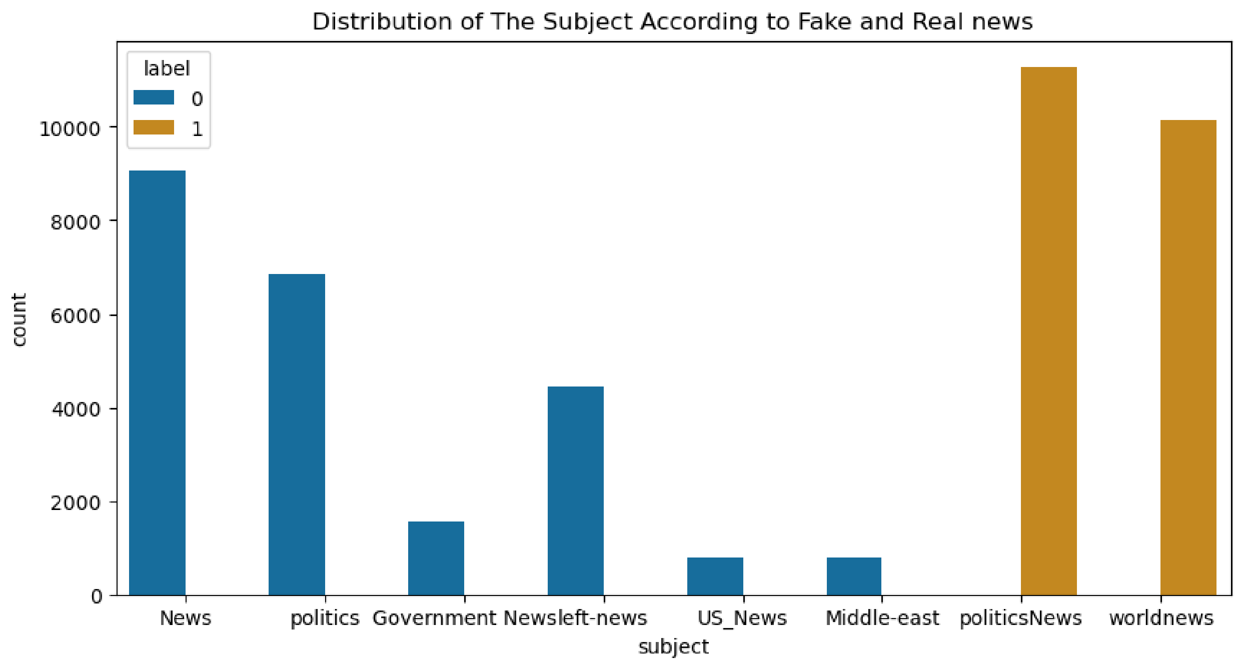

Figure 3.2.4 Distribution of The Subject According to Fake and Real news.

3.3 Data pre-processing

Data preprocessing refers to the process of cleaning, transforming, and preparing raw data for analysis. It is a critical step in the data analysis pipeline because the quality of the data used for analysis can significantly impact the accuracy and reliability of the results. During this phase, activities were carried out to construct the final dataset.

3.3.1 Data cleaning

After collecting the dataset, first we need to clean dataset. For cleaning dataset, the steps involve handling missing values, outliers, and inconsistencies in the data. Missing values can be imputed or removed based on the nature of the data and the analysis. Outliers, which are extreme values that deviate significantly from the rest of the data, can be treated by either removing them or transforming them to reduce their impact. In this case our datasets were cleaned before so we didn't need to clean it.

3.3.2 Data Integration

After that we label our datasets, “Fake.csv” we label as “0” and “real.csv” we label as “1”. And we merge our two datasets in one data frame by the help of panda library. And then we remove columns which are not required like title, subject and date.

3.3.3 Data transformation

Next, we create a function to pre-process the text data. Data often needs to be transformed to meet the assumptions of statistical tests or to improve the performance of machine learning models. Common transformations include normalization (scaling the data to a specific range), log transformation (to handle skewed data), or power transformation (to stabilize variance).

We use wordopt function to process the text data according with the following steps:

- The news article's characters are all transformed to lower case letters.
- We removed all punctuation and accent marks.
- All numerical values have been removed
- All the blank spaces have been eliminated.
- Words that are repeated are expelled.

3.3.4 Feature extraction

Selecting relevant features from the dataset is important to improve model performance and reduce computational complexity. It involves identifying and removing irrelevant or redundant features that do not contribute much to the analysis or prediction. Feature selection techniques include statistical tests, correlation analysis, or automated methods like feature importance from tree-based models. I use TF-IDF technique for feature extraction.

In the era of information overload, distinguishing between genuine and fake news has become a critical challenge. Machine learning techniques, coupled with effective feature extraction methods, have emerged as valuable tools for automating the detection of fake news. In this paper, we present a comprehensive approach that leverages the TF-IDF vectorization model to process textual data and detect fake news articles.

To effectively analyze text-based data, it is crucial to convert the textual information into numerical representations. TF-IDF vectorization, an extensively used technique in natural language processing, accomplishes this task by assigning meaningful metrics to words based on their presence and importance within the documents.

TF-IDF, an acronym for Term Frequency-Inverse Document Frequency, operates on two key principles. First, it emphasizes the importance of words that appear frequently within a document. If a word occurs multiple times in a specific document, it is likely to carry significant relevance to the content, and thus, it should be assigned higher weights. Second, TF-IDF takes into account the prevalence of a word across the entire corpus. Words that appear in numerous documents may not be unique identifiers and, therefore, should be assigned lower weights. Here is the equation for TF-IDF:

$$\text{TF-IDF} = \frac{numbers\ of\ term\ occurances}{terms\ in\ text} \times log\ \frac{number\ of\ text\ in\ collection}{number\ of\ text\ where\ term\ occurs}$$

The process of TF-IDF vectorization involves calculating the term frequency (TF) and inverse document frequency (IDF) for each word in the corpus. TF measures the frequency of a word within a particular document, while IDF quantifies how common or rare a word is across the entire collection of documents. By multiplying these two values, we obtain the TF-IDF score, which serves as a representation of the word's significance within the document and the corpus as a whole.

The inclusion of IDF in the TF-IDF vectorization process is crucial to minimize the impact of words that have high frequencies across multiple documents. These words, although commonly occurring, do not contribute substantially to the derivation of document meaning. By down weighting such terms, IDF enhances the discrimination of more unique and informative words, thereby improving the accuracy of fake news detection.

Once the textual data has been transformed into TF-IDF vectors, various machine learning algorithms can be employed for fake news detection. Decision tree classification and random forest classifiers are popular choices due to their interpretability and ability to capture complex relationships within the data. These algorithms leverage the numerical representations derived from TF-IDF to classify news articles as genuine or fake.

TF-IDF vectorization offers several advantages for fake news detection. Firstly, it enables the representation of textual data in a numerical form, allowing machine learning algorithms to process and analyze the information efficiently. Secondly, the incorporation of IDF helps prioritize unique and meaningful terms, effectively capturing the distinguishing features of fake news articles. Lastly, TF-IDF vectorization serves as a feature extraction technique that enhances the performance of machine learning models in classifying and detecting fake news.

In this paper, we have proposed a comprehensive approach for fake news detection using TF-IDF vectorization. By converting textual data into TF-IDF vectors, we effectively capture the significance of words within documents and the entire corpus. The incorporation of IDF ensures that common words do not overshadow distinctive terms, enhancing the accuracy of fake news detection. Through the utilization of machine learning algorithms, such as decision tree classification and random forest classifiers, we can effectively classify and identify fake news articles. The combination of TF-IDF vectorization and machine learning algorithms offers a powerful approach to automate the detection of fake news, contributing to the development of robust and reliable systems in combating misinformation.

3.3.5 Train test split

Train-test split is a common technique used in machine learning to evaluate the performance of a model. The idea is to split the available data into two sets: a training set and a test set. The training set is used to train the model, and the test set is used to evaluate its performance on new, unseen data. The goal is to develop a model that generalizes well to new data, rather than simply memorizing the training data. Its basic syntax follows the pattern: train-test-split(X, y, train_size=0.0, test_size=0.0, random_state=None).

Let's take a closer look at the parameters of this function. The "X" variable typically represents the input features of the dataset, while "y" corresponds to the target variable or labels. These variables allow us to separate the data for training and testing purposes effectively.

The train_size parameter determines the size of the training dataset. By default, it is set to None, but it can be assigned an integer value if you want an exact number of samples or a float value between 0.1 and 1.0 to indicate a proportion of the data.

Similarly, the test_size parameter controls the size of the testing dataset. If train_size is set to None, the default value of test_size is 0.25 to complement the split. Like train_size, it can be specified as an integer or a float between 0.1 and 1.0.

The random_state parameter is used to control the shuffling of the data during the splitting process. Setting it to an integer value allows you to reproduce the same split across multiple function calls. If not specified, it defaults to None, which results in a random shuffling.

Typically, the data is split randomly into a training set (usually 70-80% of the data) and a test set (usually 20-30% of the data). In our case we used 80% data for training and 20% data for testing. We then trained our model on the training set and then evaluated on the test set. The performance on the test set is used to estimate how well the model performed on new, unseen data. It's important to ensure that the test set is representative of the data as a whole. This means that the test set should have a similar distribution of data to the training set. If the test set is too different from the training set, the model may not generalize well to new data.

In addition to a train-test split, there are other techniques for evaluating the performance of a model, such as cross-validation, which involves splitting the data into multiple folds and using each fold as a test set while training on the remaining folds.

3.4 Machine learning models

After splitting our dataset into training and testing we use five different machine learning algorithm to find an algorithm which one will fit in our model perfectly and accurately. Model with good f1 score and confusion matrix can better perform. Let’s talk about this algorithm and how they work.

3.4.1 Decision Tree Classification

Decision tree classification is a popular machine learning algorithm used for both classification and regression tasks. In decision tree classification, a tree-like model is constructed that makes a sequence of decisions based on the features of the input data to classify the data into one or more classes.

The decision tree is built using a training dataset, where each data point is represented by a set of features and a corresponding class label. The algorithm works by recursively splitting the data into smaller subsets based on the values of the features. The splitting is done in such a way that it maximizes the separation between different classes.

At each node of the tree, a decision is made based on the value of one of the features. This decision splits the data into two or more branches, each corresponding to a different value of the feature. The process continues until the data at each leaf node belongs to a single class.

Once the decision tree is constructed, it can be used to classify new data points by following the decision path from the root node to a leaf node. At each node, the value of the corresponding feature is checked, and the algorithm moves to the appropriate child node based on the value of the feature. The process continues until a leaf node is reached, and the class label associated with that leaf node is returned as the predicted class label for the input data point.

Decision tree classification is a popular algorithm because it is easy to interpret and can handle both categorical and numerical data. However, it can be prone to overfitting, wherethe tree is overly complex and fits the training data too closely, resulting in poor generalization to new data. To prevent overfitting, techniques such as pruning, limiting the depth of the tree, and using ensemble methods such as random forests can be employed.

Overall, decision tree classification is a powerful and widely used algorithm in machine learning, with applications in a variety of domains such as healthcare, finance, and marketing.

3.4.2 Logistic Regression

Logistic Regression is a popular machine learning algorithm used for classification tasks. Unlike linear regression, which is used for predicting continuous values, logistic regression predicts the probability of an input belonging to a certain class.

In logistic regression, the input data is a set of features, and the output is a binary class label (0 or 1). The algorithm works by modeling the probability of the input belonging to the positive class (class 1) as a function of the input features. This function is called the logistic function, which maps any real-valued input to a value between 0 and 1:

P(y=1|x) = 1 / (1 + exp(-z))

where z is a linear combination of the input features:

z = b0 + b1x1 + b2x2 + ... + bn*xn

Here, b0, b1, b2, ..., bn are the coefficients of the logistic regression model that are learned during training. The logistic function is also called the sigmoid function.

During training, the algorithm adjusts the coefficients of the logistic regression model to minimize the difference between the predicted probability and the actual class labels. This is typically done using maximum likelihood estimation or gradient descent.

Once the model is trained, it can be used to classify new input data by using the logistic function to compute the probability of the input belonging to the positive class. If the probability is greater than a threshold (usually 0.5), the input is classified as belonging tothe positive class, otherwise, it is classified as belonging to the negative class.

Logistic regression is a powerful and widely used algorithm in machine learning due to several reasons. First, it is simple and easy to interpret, making it a popular choice for many applications. Second, it can handle both categorical and numerical input data. Third, it can be extended to handle multiclass classification tasks using techniques such as one-vs-rest or multinomial logistic regression.

However, logistic regression has some limitations. It assumes that the relationship between the input features and the output is linear, which may not always be the case. It also assumes that the input features are independent, which may not be true in many real-world applications. Additionally, logistic regression can suffer from overfitting if the model is too complex or if the input data has a large number of features.

Overall, logistic regression is a powerful and widely used algorithm in machine learning with applications in various domains such as healthcare, finance, and marketing.

3.4.3 Passive Aggressive Classifier

The Passive Aggressive (PA) classifier is a machine learning algorithm used for binary classification tasks. It is a variant of the perceptron algorithm that is designed to handle online learning, where the model is updated incrementally as new data becomes available.

In the PA algorithm, the input data is a set of features, and the output is a binary class label (0 or 1). The algorithm works by maintaining a weight vector that is used to compute a score for each input data point. The sign of the score is used to predict the class label:

$$y = \mathrm{sign}(w \cdot x)$$

where w is the weight vector, x is the input feature vector, and sign() is the sign function.

During training, the PA algorithm updates the weight vector based on the difference between the predicted class label and the true class label. The update rule is designed to minimize the loss function while ensuring that the weight vector remains close to the current estimate. The update rule is given by:

$$w = w + \eta * \max(0, 1 - y(w \cdot x)) * y * x$$

where η is the learning rate, y is the true class label, and the max() function ensures that the update is only performed if the prediction is incorrect.

The PA algorithm is called "passive-aggressive" because it is passive when the prediction is correct and aggressive when the prediction is incorrect. This allows the algorithm to quickly adapt to changes in the input data and handle non-stationary environmentswhere the underlying data distribution can change over time.

The PA algorithm has several advantages over other classification algorithms. First, it is computationally efficient and can handle large-scale datasets. Second, it can handle non-linearly separable data using kernel methods. Third, it can be easily extended to handle multiclass classification tasks using techniques such as one-vs-all or one-vs-one.

However, the PA algorithm also has some limitations. It can suffer from overfitting if the learning rate is too high or if the input data is noisy. It can also be sensitive to the choice of regularization parameter and may require tuning for optimal performance.

Overall, the Passive Aggressive classifier is a powerful and widely used algorithm in machine learning with applications in various domains such as text classification, image classification, and anomaly detection.

3.4.4 Gradient Boosting

Gradient Boosting is a popular machine learning algorithm used for both regression and classification tasks. It is a type of ensemble learning algorithm that combines multiple weak models, such as decision trees, to create a strong predictive model.

In Gradient Boosting, the algorithm works by iteratively adding new models to the ensemble, with each new model correcting the errors made by the previous models. The algorithm starts by training a single base model, such as a decision tree, on the training data. The base model makes predictions on the input data, and the errors are computed as the difference between the predicted values and the true values.

In the next iteration, a new model is added to the ensemble, which is trained on the errors made by the previous model. The new model is trained to predict the errors made by the previous model, rather than the true values. The predictions of the new model are then added to the predictions of the previous model, and this process is repeated until a stopping criterion is met.

The key idea behind Gradient Boosting is to use gradient descent to minimize a loss function, such as mean squared error or cross-entropy, that measures the difference between the predicted values and the true values. During each iteration, the algorithm computes the gradient of the loss function with respect to the predictions of the previous model. This gradient is used to update the parameters of the new model, such as the split points in a decision tree.

The final prediction of the ensemble is the sum of the predictions of all themodels in the ensemble. This is equivalent to computing a weighted average of the predictions, where the weights are determined by the performance of each model on the training data.

Gradient Boosting has several advantages over other machine learning algorithms. First, it can handle both categorical and numerical data. Second, it can handle missing values in the input data. Third, it can capture non-linear relationships between the input features and the output variable.

However, Gradient Boosting also has some limitations. It can be sensitive to the choice of hyperparameters, such as the learning rate and the number of iterations. It can also suffer from overfitting if the model is too complex or if the input data has a large number of features. Additionally, Gradient Boosting can be computationally expensive and may require a large amount of memory to store the ensemble of models.

Overall, Gradient Boosting is a powerful and widely used algorithm in machine learning with applications in various domains such as healthcare, finance, and marketing. It is particularly useful when high accuracy is required, and the input data has complex relationships between the features and the output variable.

3.4.5 Random Forests Classifier

Random Forests is a popular machine learning algorithm used for classification and regression tasks. It is a type of ensemble learning algorithm that combines multiple decision trees to create a strong predictive model.

In Random Forests, the algorithm works by building a set of decision trees on random subsets of the input data and features. The decision trees are trained independently on different subsets of the data, where each tree is trained on a different subset of the features, selected randomly. This helps to reduce overfitting and improve generalization performance.

During training, the algorithm selects a random subset of the data and features for each tree and grows the tree using a recursive partitioning algorithm. At each node of the tree, the algorithm selects the best feature to split the data based on some criterion, such as information gain or Gini impurity. The process is repeated until the tree reaches a maximum depth or the number of nodes reaches a pre-defined limit.

Once all the trees are trained, they are combined to make predictions on new input data. The final prediction is the class label that is most frequently predicted by the individual trees in the forest.

Random Forests has several advantages over other machine learning algorithms. First, it can handle both categorical and numerical data. Second, it can handle missing values in the input data. Third, it can capture non-linear relationships between the input features and the output variable.

Random Forests also has some limitations. It can be sensitive to the choice of hyperparameters, such as the number of trees in the forest, the maximum depth of the trees, and the size of the random subsets used for training. It can also be computationally expensive, especially for large datasets or when the number of trees in the forest is high.

Overall, Random Forests is a powerful and widely used algorithm in machine learning with applications in various domains such as healthcare, finance, and marketing. It is particularly useful when high accuracy is required, and the input data has complex relationships between the features and the output variable. It is also useful when the input data has noise or missing values, as Random Forests can handle these issues effectively.

3.5 Model evaluation and comparison

Model evaluation and comparison are important steps in machine learning to assess the performance of different machine learning algorithms and choose the best model for a given problem. There are several metrics and techniques that can be used for model evaluation and comparison. Here are few commonly used techniques:

Splitting the data: Dividing dataset into two or three subsets: training set, validation set, and test set. The training set is used to train the model, the validation set is used for hyperparameter tuning and model selection, and the test set is used to evaluate the final performance of the selected model.

Choosing evaluation metrics: Select appropriate evaluation metrics based on the problem type. For classification tasks, common metrics include accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC). For regression tasks, metrics like mean squared error (MSE), mean absolute error (MAE), and R-squared are commonly used. Choose metrics that align with the specific goals of your problem.

Accuracy: Accuracy is a common metric used to evaluate classification algorithms. It measures the proportion of correctly classified instances in the test set. However, accuracy can be misleading if the class distribution is imbalanced or the cost of misclassifying the positive and negative classes is different.

Precision and recall: Precision and recall are two metrics used to evaluate the performance of a classification algorithm in the presence of class imbalance. Precision measures the proportion of true positive classifications among all positive classifications, while recall measures the proportion of true positive classifications among all actual positive instances. The F1 score is a harmonic mean of precision and recall and is often used as a single metric to evaluate classification algorithms.

Mean Squared Error (MSE): MSE is a common metric used to evaluate regression algorithms. It measures the average squared difference between the predicted and actual values. The lower the MSE, the better the algorithm's performance.

R-squared: R-squared is another metric used to evaluate regression algorithms. It measures the proportion of variance in the dependent variable that is explained by the independent variables. R-squared values range from 0 to 1, with higher values indicating better performance.

Cross-validation: Cross-validation is a technique used to evaluate the performance of a machine learning algorithm by dividing the dataset into multiple subsets or folds. The algorithm is trained on a subset of the data and evaluated on the remaining subset. This process is repeated multiple times, with each subset serving as the test set once. The average performance across all the folds is used as the final performance metric. Cross-validation is useful for estimating the generalization performance of a model and for selecting hyperparameters that can be tuned to improve performance.

Training and tuning models: Train different models on the training set using various algorithms and hyperparameter configurations. Perform cross-validation on the training set or use the validation set to tune hyperparameters and select the best-performing model based on the chosen evaluation metrics.

Evaluating models: Evaluate the performance of each model on the test set using the selected evaluation metrics. Compare the results of different models to determine which one performs better.

Model evaluation and comparison are important steps in machine learning. The choice of metrics and techniques used for evaluation and comparison depends on the type of problem and the data. It is important to select relevant metrics and techniques that provide a comprehensive evaluation of the model's performance.

CHAPTER 4

EXPERIMENTS AND RESULTS

4.1 Experimental setup

To do our experiment we use a laptop with core i3 processor, 8GB ram and 512 GB SSD. We use Jupyter Notebook for programming platform and Python as a programming language.

Jupyter Nootbook: Jupyter Notebook is an open-source web application that allows users to create and share documents that contain live code, equations, visualizations, and narrative text. It is widely used in data science and scientific computing as a tool for interactive coding, data exploration, and collaboration. Jupyter Notebook supports multiple programming languages, including Python, R, and Julia, and provides a rich set of features such as code highlighting, inline plotting, and the ability to run code cells in any order. It also allows users to export the notebook in various formats such as HTML, PDF, and Markdown. Overall, Jupyter Notebook is a powerful tool for data analysis, visualization, and machine learning.

Python Programming: Python is a high-level, interpreted programming language that is easy to learn and widely used for a variety of applications, including web development, scientific computing, data analysis, and artificial intelligence. It has a simple syntax and a large standard library, as well as a thriving ecosystem of third-party packages and tools. Python's popularity is due in part to its readability and ease of use, as well as its versatility and flexibility. It is open-source software, meaning that it is free to use and distribute, and can be run on a variety of platforms, including Windows, Linux, and macOS. We use Python Programming Language for coding and building our machine learning model on Windows machine. Here are important libraries we use:

- Numpy
  NumPy is a Python library for numerical computing that provides support for large, multi-dimensional arrays and matrices, as well as a collection of mathematical functions to operate on these arrays. It is widely used in scientific computing, data analysis, and machine learning and is known for its efficiency and ease of use.

- Pandas
  Pandas is a Python library used for data manipulation and analysis. It provides data structures for efficiently storing and manipulating large datasets, as well as a wide range of tools for data cleaning, transformation, and visualization.

- SciKit-learn
  Scikit-learn is a popular Python library for machine learning that provides a range of tools for classification, regression, clustering, and dimensionality reduction. It includes a wide range of algorithms for machine learning tasks, as well as tools for model selection, preprocessing, and evaluation. Scikit-learn is widely used in academic and industrial

settings for developing and deploying machine learning applications and is known for its flexibility, ease of use, and integration with other Python libraries.

- Seaborn
  Seaborn is a Python data visualization library based on Matplotlib that provides a high-level interface for creating informative and attractive statistical graphics.

- Matplotlib
  Matplotlib is a Python library used for creating static, animated, and interactive visualizations in Python. It provides a wide range of charts, plots, and graphs for representing data in a variety of formats and is widely used in scientific computing, data analysis, and machine learning. Matplotlib is known for its flexibility and ease of use, and is highly customizable, allowing users to create visually appealing and informative visualizations for their data.

- String
  The string library in Python provides a collection of constants and functions for working with strings. It includes constants for ASCII letters, digits, and punctuation, as well as functions for formatting, encoding, and decoding strings.

- Re
  The re library in Python provides support for regular expressions and pattern matching. It includes functions for searching, replacing, and manipulating strings using regular expressions, which are powerful tools for text processing and data cleaning.

4.2 Results and analysis

We use five different Machine learning models for detecting fake news. The five different models are Decision Tree Classifier (DTC), Logistic Regression (LR), Passive Aggressive Classifier (PAC), Gradient Boosting Classifier (GBC), Random Forest Classifier (RFC).

4.2.1 DecisionTreeClassifier (DTC)

The model achieved an overall accuracy of 99.57%, which indicates that the model is highly accurate in predicting whether a news article is real or fake.

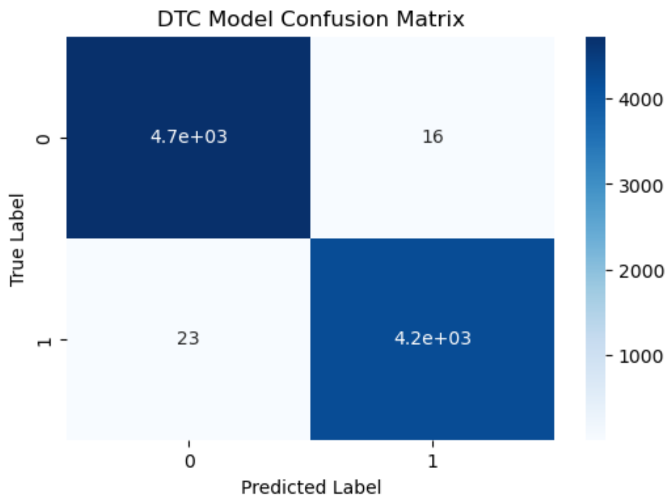

Figure 4.1 DTC Confusion Matrix

The confusion matrix represents the performance of Decision Tree Classifier (DTC) model, for fake news detection. The matrix is a 2x2 table that shows the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) predicted by the model.

In this confusion matrix, the model correctly predicted 4723 true negatives and 4218 true positives, while incorrectly predicting 16 false positives and 23 false negatives. The true positives and true negatives represent the number of correctly predicted real and fake news articles, respectively, while the false positives and false negatives represent the number of misclassified articles.

The matrix shows that the model has a high accuracy and precision, as the majority of the predictions are true positives and true negatives. The small number of false positives and false negatives indicates that the model is performing well in differentiating between real and fake news articles.

Overall, the confusion matrix suggests that the Decision Tree Classifier model is highly accurate and reliable for fake news detection, with a high true positive rate and a low false positive rate.

	precision	recall	f1-score	support
0	1	1	1	4739
1	1	0.99	1	4241
accuracy			1	8980
macro avg	1	1	1	8980
weighted avg	1	1	1	8980

Table 4.1 DTC Classification Report

The classification report result shows the performance of a Decision Tree Classifier machine learning model for fake news detection. The precision and recall scores for both classes (real and fake) are 1.00 and 0.99, indicating that the model is correctly identifying all instances of real and fake news. The F1-score, which is the harmonic mean of precision and recall, is also 1.00 for both classes, indicating that the model is equally good at identifying both real and fake news. The macro average, which takes the mean of precision, recall, and F1-score across classes, is also 1.00, indicating high performance across all classes. Similarly, the weighted average, which considers class imbalance, is 1.00, suggesting that the model handles different class sizes effectively.

Overall, this classification report demonstrates that the Decision Tree classifier model is highly accurate and performs well for fake news detection.

4.2.2 LogisticRegression (LR)

The model achieved an overall accuracy of 98.39%, which indicates that the model is good enough in predicting whether a news article is real or fake.

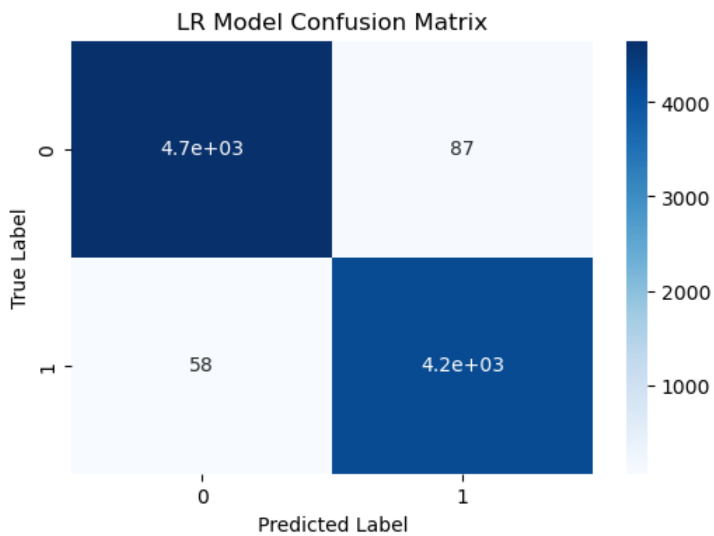

Figure 4.2 LR Confusion Matrix

The confusion matrix represents the performance of logistic regression model, for fake news detection. In this confusion matrix, the model correctly predicted 4652 true negatives and 4183 true positives, while incorrectly predicting 87 false positives and 58 false negatives. The true positives and true negatives represent the number of correctly predicted real and fake news articles, respectively, while the false positives and false negatives represent the number of misclassified articles.

The matrix shows that the model has a high accuracy and precision, as the majority of the predictions are true positives and true negatives. However, the number of false positives and false negatives indicates that the model is not perfect and there is potential for improvement.

	precision	recall	f1-score	support
0	0.99	0.98	0.98	4739
1	0.98	0.99	0.98	4241
accuracy			0.98	8980
macro avg	0.98	0.98	0.98	8980
weighted avg	0.98	0.98	0.98	8980

Table 4.2 LR Classification report

The classification report shows the performance of a Logistic Regression (LR) machine learning model for fake news detection. The precision score for class 0 (fake) is 0.99 and for class 1 (real) is 0.98. And the recall score for class 0 (fake) is 0.98 and for class 1 (real) is 0.99, indicating that the model is correctly identifying most of the instances of real and fake news. The F1-score, which is the harmonic mean of precision and recall, is 0.98 for both classes, indicating that the model is equally good at identifying both real and fake news.

4.2.3 PassiveAggressiveClassifier (PAC)

The model achieved an overall accuracy of 99.39%, which indicates that the model is highly accurate in predicting whether a news article is real or fake.

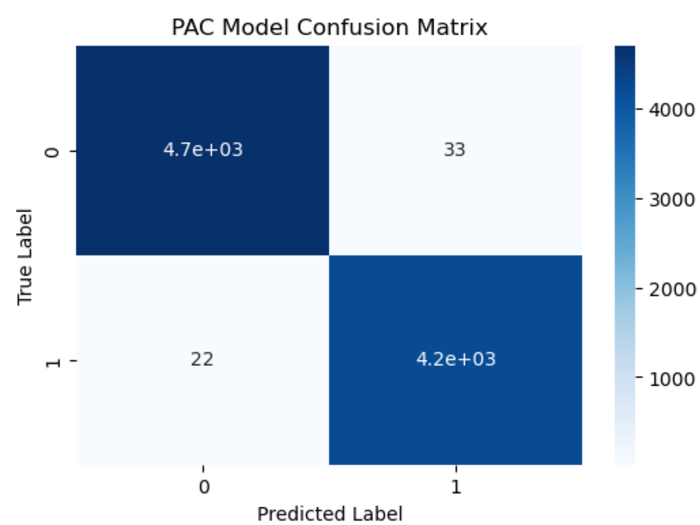

Figure 4.3 PAC Confusion Matrix

The confusion matrix represents the performance of a machine learning model, in this case, a Passive Aggressive Classifier (PAC) model, for fake news detection. The matrix is a 2x2 table that shows the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) predicted by the model.

In this confusion matrix, the model correctly predicted 4706 true negatives and 4219 true positives, while incorrectly predicting 33 false positives and 22 false negatives. The true positives and true negatives represent the number of correctly predicted real and fake news articles, respectively, while the false positives and false negatives represent the number of misclassified articles.

The matrix shows that the model has a high accuracy and precision, as the majority of the predictions are true positives and true negatives. The small number of false positives and false negatives indicates that the model is performing well in differentiating between real and fake news articles.

Overall, the confusion matrix suggests that the Passive Aggressive Classifier model is highly accurate and reliable for fake news detection, with a high true positive rate and a low false positive rate.

	precision	recall	f1-score	support
0	1	0.99	0.99	4739
1	0.99	0.99	0.99	4241
accuracy			0.99	8980
macro avg	0.99	0.99	0.99	8980
weighted avg	0.99	0.99	0.99	8980

Table 4.3 PAC Classification Report

The classification report shows the performance of a Passive Aggressive Classifier (PAC) machine learning model for fake news detection. The precision score for class 0 (fake) is 1and for class 1 (real) is 0.99 and the recall scores for both classes (real and fake) are 0.99, indicating that the model is correctly identifying most of the instances of real and fake news. The F1-score, which is the harmonic mean of precision and recall, is also 0.99 for both classes, indicating that the model is equally good at identifying both real and fake news.

4.2.4 GradientBoostingClassifier (GBC)

The model achieved an overall accuracy of 99.44%, that means the model is accurate in predicting whether a news article is real or fake.

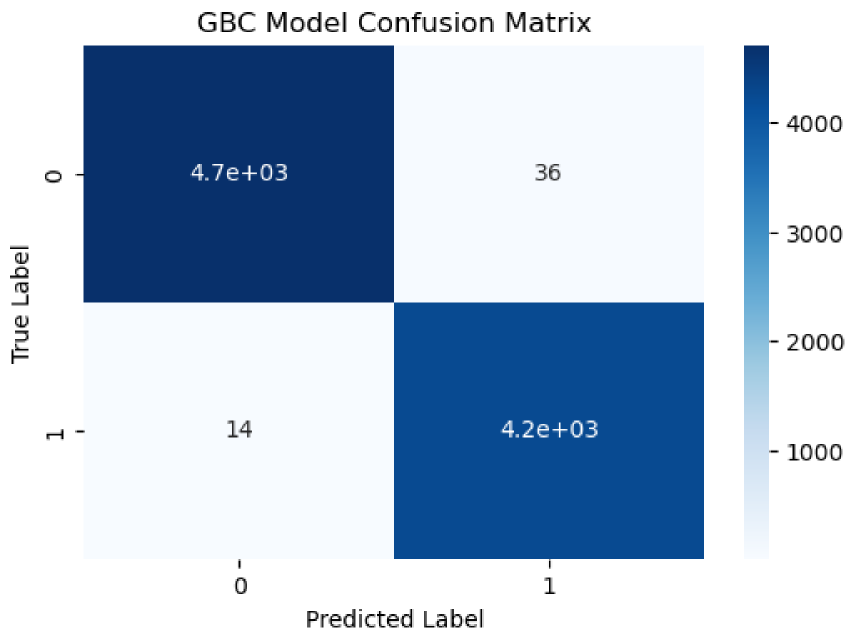

Figure 4.4 GBC Confusion Matrix

The confusion matrix represents the performance of Gradient Boosting Classifier (GBC) model, for fake news detection. In this confusion matrix, the model correctly predicted 4703 true negatives and 4227 true positives, while incorrectly predicting 36 false positives and 14 false negatives. The true positives and true negatives represent the number of correctly predicted real and fake news articles, respectively, while the false positives and false negatives represent the number of misclassified articles.

The matrix shows that the model has a high accuracy and precision, as the majority of the predictions are true positives and true negatives. The small number of false positives and false negatives indicates that the model is performing well in differentiating between real and fake news articles. Overall, the confusion matrix suggests that the Gradient Boosting Classifier model is highly accurate and reliable for fake news detection, with a high true positive rate and a low false positive rate. Therefore, further testing and evaluation of the model may be necessary to ensure its generalizability and robustness.

	precision	recall	f1-score	support
0	1	0.99	0.99	4739
1	0.99	1	0.99	4241
accuracy			0.99	8980
macro avg	0.99	0.99	0.99	8980
weighted avg	0.99	0.99	0.99	8980

Table 4.4 GBC Classification Report

The classification report shows the performance of a Gradient Boosting Classifier (GBC) machine learning model for fake news detection. The precision and recall scores for both classes (real and fake) are 1.00 and 0.99, respectively, indicating that the model is correctly identifying most of the instances of real and fake news. The F1-score, which is the harmonic mean of precision and recall, is 0.99 for real and fake news, indicating that the model is performing very well in identifying real news and slightly less well for fake news.

4.2.5 RandomForestClassifier (RFC)

The model achieved an overall accuracy of 99.09%, that means the model is accurate in predicting whether a news article is real or fake.

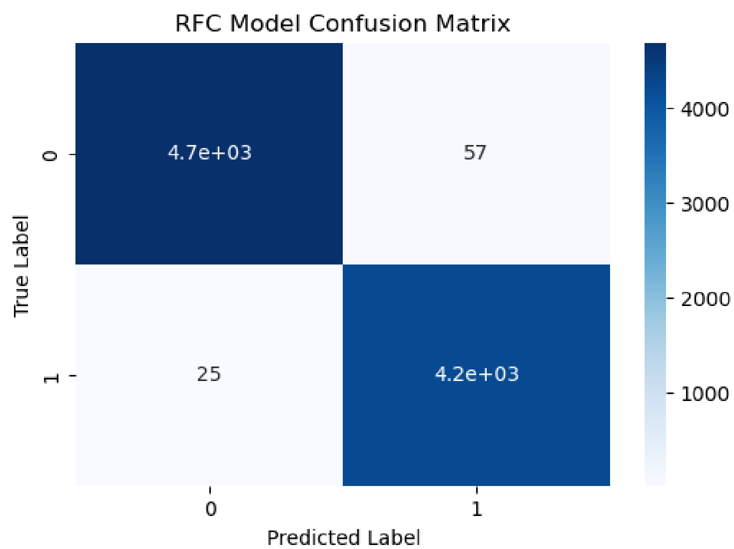

Figure 4.5 RFC Confusion Matrix

The confusion matrix represents the performance of Random Forest Classifier (RFC) model, for fake news detection. In this confusion matrix, the model correctly predicted 4682 true negatives and 4216 true positives, while incorrectly predicting 57 false positives and 25 false negatives. The true positives and true negatives represent the number of correctly predicted real and fake news articles, respectively, while the false positives and false negatives represent the number of misclassified articles.

The matrix shows that the model has a high accuracy and precision, as the majority of the predictions are true positives and true negatives. However, the number of false positives and false negatives indicates that the model is not perfect and there is potential for improvement.

	precision	recall	f1-score	support
0	0.99	0.99	0.99	4739
1	0.99	0.99	0.99	4241
accuracy			0.99	8980
macro avg	0.99	0.99	0.99	8980
weighted avg	0.99	0.99	0.99	8980

Table 4.5 RFC Classification Matrix

The classification report shows the performance of a Random Forest Classifier (RBC) machine learning model for fake news detection. The precision and recall scores for both classes (real and fake) are 0.99, indicating that the model is correctly identifying most of the instances of real and fake news. The F1-score, which is the harmonic mean of precision and recall, is also 0.99 for both classes, indicating that the model is equally good at identifying both real and fake news.

4.3 Comparison of different models

Model	Accuracy	Class	Precision	Recall	F1-score	Support
DTC	99.57	F R	1.00 1.00	1.00 0.99	1.00 1.00	4739 4241
LR	98.39	F R	0.99 0.98	0.98 0.99	0.98 0.98	4739 4241
PAC	99.39	F R	1.00 0.99	0.99 0.99	0.99 0.99	4739 4241
GBC	99.44	F R	1.00 0.99	0.99 1.00	0.99 0.99	4739 4241
RFC	99.09	F R	0.99 0.99	0.99 0.99	0.99 0.99	4739 4241

Table 4.6 Comparison of Different Model’s Accuracy, Precision, Recall, F1-score

The Decision Tree Classifier (DTC) achieves the highest accuracy of 99.57%, indicating that it correctly classifies the majority of the instances in the dataset. The precision score of 1.00 suggests that when the DTC predicts a positive class (F), it is almost always correct. Similarly, the recall score of 1.00 indicates that the DTC can identify almost all positive instances correctly. The F1-score of 0.99 indicates a balance between precision and recall.

The Logistic Regression (LR) model performs slightly lower than the DTC. It has an accuracy of 98.39%, indicating that it correctly classifies a significant portion of the dataset. The precision score of 0.99 indicates that the LR model has a high proportion of correct predictions for the positive class. The recall score of 0.98 suggests that the LR model identifies most positive instances but might miss a small portion of them. The F1-score of 0.98 indicates a good balance between precision and recall but slightly lower performance compared to the DTC.

The Passive Aggressive Classifier (PAC) achieves an accuracy of 99.39%, indicating high overall performance. The precision score of 1.00 suggests that the PAC model rarely predicts positive instances incorrectly. The recall score of 0.99 indicates that the PAC model can correctly identify most positive instances, but it might miss a few. The F1-score of 0.99 indicates a good balance between precision and recall.

The Gradient Boosting Classifier (GBC) achieves an accuracy of 99.44%, which is very close to the highest performing model, DTC. The precision score of 1.00 indicates that the GBC model has a high proportion of correct predictions for the positive class. The recall score of 0.99 suggests that the GBC model can correctly identify most positive instances but might miss a few. The F1-score of 0.99 indicates a good balance between precision and recall.

The Random Forest Classifier (RFC) achieves an accuracy of 99.09%, slightly lower than the top-performing models. The precision score of 0.99 indicates that the RFC model has a high proportion of correct predictions for the positive class. The recall score of 0.99 suggests that the RFC model can correctly identify most positive instances. The F1-score of 0.99 indicates a good balance between precision and recall.

Overall, all the models perform well with high accuracy and similar precision, recall, and F1-scores. The DTC model stands out with the highest accuracy, precision, and recall, making it the best-performing model.

CHAPTER 5

DISCUSSION

5.1 Error analysis and model interpret

To analyze the errors and interpret the models, we can start by looking at the confusion matrices for each model. The confusion matrix shows the counts of true negatives (TN), false positives (FP), false negatives (FN), and true positives (TP). Based on the provided confusion matrices for different models (DTC, LR, PAC, GBC, RFC), here's the analysis:

Model	True Negative	False Positive	False Negative	True Positive
DTC	4723	16	23	4218
LR	4652	87	58	4183
PAC	4706	33	22	4219
GBC	4703	36	14	4227
RFC	4682	57	25	4216

Table 4.6 Error Analysis

The DTC model has a high number of true negatives (correctly classified negatives) and true positives (correctly classified positives), indicating good performance in identifying both classes. It has a low number of false positives and false negatives, suggesting a relatively balanced and accurate model.

The LR model also has a high number of true negatives and true positives, indicating good performance in classification.
However, it has a higher number of false positives and false negatives compared to DTC, suggesting that it may be more prone to misclassifications.

The PAC model has a similar performance to DTC, with high numbers of true negatives and true positives.
It has a low number of false positives and false negatives, indicating good accuracy in classification.

The GBC model performs similarly to DTC and PAC, with high numbers of true negatives and true positives.
It has a slightly higher number of false positives compared to the previous models but maintains a low number of false negatives.

The RFC model performs well with high numbers of true negatives and true positives.
It has a slightly higher number of false positives compared to DTC and PAC but still maintains a relatively low number of false negatives.

Based on the provided information, all the models (DTC, LR, PAC, GBC, RFC) exhibit relatively good performance with high true negative and true positive counts. However, the presence of false positives and false negatives indicates areas where the models can be improved. Further analysis can be performed to understand the factors contributing to these errors, such as examining misclassified examples, feature importance, or exploring the data distribution.

5.2 Implications and mitigation.

Fake news detection and mitigation are critical in today's information landscape, where the rapid spread of misinformation can have serious consequences. Here are some implications for fake news detection and mitigation:

Advancements in Natural Language Processing (NLP): NLP techniques, including machine learning and deep learning, play a vital role in fake news detection. As these technologies continue to evolve, we can expect improved accuracy in identifying misleading or fabricated content. NLP models can analyze text patterns, linguistic features, and contextual cues to flag potential fake news articles.

Automated Fact-Checking Systems: With the help of AI, automated fact-checking systems are being developed to quickly verify claims made in news articles. These systems can analyze large amounts of data and compare statements against trusted sources to determine their accuracy. By automating fact-checking processes, it becomes easier to identify and debunk fake news stories at scale.

Collaborative Efforts: Addressing fake news requires collaboration between technology companies, news organizations, fact-checkers, and policymakers. By working together, these stakeholders can share resources, develop common standards, and create effective strategies for identifying and countering fake news. Collaboration can also help in establishing reliable platforms for users to access accurate information.

Social Media Policies and Algorithms: Social media platforms have a significant influence on the spread of fake news due to their large user bases and algorithmic content distribution. Platforms have started implementing stricter policies to curb the dissemination of misinformation. These policies may involve fact-checking labels, warning prompts, or even removing false content altogether. Ongoing efforts to refine algorithms can prioritize trustworthy sources and reduce the visibility of misleading information.

Media Literacy and Education: Improving media literacy is essential for individuals to discern reliable information from fake news. Education programs can equip people with critical thinking skills and teach them how to evaluate sources, cross-reference information, and identify common tactics used in spreading misinformation. By promoting media literacy, individuals become more resilient to fake news and can actively participate in mitigating its impact.

Ethical AI Development: Developers and researchers have a responsibility to build AI systems that prioritize accuracy, fairness, and transparency in fake news detection. Bias mitigation techniques should be employed to ensure that these systems do not inadvertently discriminate against certain perspectives or amplify existing biases. Striving for ethical AI development helps maintain trust in the technology and its applications.

Overall, the implications for fake news detection and mitigation involve advancements in technology, collaborative efforts, policy changes, and promoting media literacy. By combining these approaches, we can make significant progress in combating the spread of fake news and fostering a more informed society.

5.3 Limitations and future work.

Fake news detection is a challenging and important problem, but it also has several limitations and opportunities for future work. Here are a few potential limitations and avenues for future work:

Limited scope: Many current fake news detection models focus on detecting fake news articles, but may not be able to detect fake news spread through other mediums, such as social media posts, images, or videos. Future work could focus on developing models that can detect fake news across a wider range of mediums and formats.

Data bias: Fake news detection models may be prone to biases in the training data used to train them. For example, if the training data is biased towards certain sources or political views, the model may struggle to detect fake news from other sources or with different political views. Future work could focus on developing methods for detecting and mitigating biases in training data.

Adversarial attacks: Fake news producers may attempt to evade detection by using adversarial techniques, such as modifying the content or style of the fake news. Future work could focus on developing models that are robust to these types of attacks.

Multilingual support: Many fake news detection models are designed to work with English-language text and may not be able to detectfake news in other languages. Future work could focus on developing models that can detect fake news in multiple languages, particularly in regions where multiple languages are spoken.

Real-time detection: Fake news can spread rapidly on social media and other platforms, making real-time detection crucial. Future work could focus on developing models that can detect fake news in real-time, potentially using techniques such as stream processing or online learning.

Human-in-the-loop approaches: While machine learning models can be effective for detecting fake news, they may not be perfect and may generate false positives or false negatives. Future work could focus on developing human-in-the-loop approaches that combine the strengths of machine learning models with human expertise and oversight to improve the accuracy and reliability of fake news detection.

Collaboration and benchmarking: Finally, improving the state of the art in fake news detection will require collaboration and benchmarking across the research community. Future work could focus on developing standardized datasets and evaluation metrics, as well as fostering collaboration and sharing of models and techniques across researchers and organizations.

Overall, there are many exciting opportunities for future work in fake news detection, and addressing these limitations will be crucial for improving the accuracy and reliability of these important tools.

CONCLUTION

In conclusion, the proliferation of fake news in the digital age poses a significant threat to society, requiring effective detection mechanisms to preserve the integrity of information and promote informed decision-making. This research has demonstrated the potential of machine learning techniques in detecting fake news articles on media platforms.

Through the structured methodology of text pre-processing, feature extraction, and fake news detection, the study achieved impressive results using a diverse dataset and five different supervised machine learning algorithms. The Decision tree classifier emerged as the most successful model, attaining an accuracy rate of 99.57%. The research findings highlight the efficacy of the proposed technique in surpassing previous state-of-the-art results and improving the identification of false news.

Furthermore, the comprehensive literature review conducted as part of this research emphasized the challenges and limitations faced by current fake news detection approaches. The study reinforced the importance of feature selection, parameter optimization, and the development of reliable performance metrics to enhance the accuracy and interpretability of fake news detection models.

The research also acknowledged the existence of fake news variants and recognized the potential of combining a hybrid model with a large dataset to detect fake news more effectively. Notably, the study identified the need for future exploration of algorithms capable of detecting misinformation in video and image-based mediums, which are frequently exploited for spreading fake news.

Ultimately, this research contributes to the development of accurate and reliable methods for detecting fake news, offering a potential solution to address the challenges posed by misinformation. By leveraging machine learning techniques, the study aims to create a more trustworthy media landscape, empowering individuals and society with reliable information for informed decision-making.

To combat the pervasive issue of fake news, further research and innovation are required. Future studies could explore advanced machine learning algorithms, alternative feature extraction techniques, and the integration of multi-modal data to improve the detection accuracy of fake news across various mediums. Additionally, collaboration between researchers, media organizations, and policymakers is essential to develop robust frameworks that promote responsible information dissemination and educate the public on identifying and combating fake news effectively.

By leveraging the power of machine learning, society can tackle the menace of fake news and foster a more informed, responsible, and trustworthy information ecosystem for the betterment of all.

REFERENCES

1. News Use Across Social Media Platforms 2017. [online]. Available: https://www.pewresearch.org/journalism/2017/09/07/news-use-across-social-media-platforms-2017/
2. Why we fall for fake news: Hijacked thinking or laziness? [online]. Available: https://www.apa.org/news/apa/2020/fake-news
3. Many Americans Believe Fake News Is Sowing Confusion. [online] Available: https://www.pewresearch.org/journalism/2016/12/15/many-americans-believe-fake-news-is-sowing-confusion/
4. Fake News Is Poisoning Brazilian Politics. WhatsApp Can Stop It. [online]. Available: https://www.nytimes.com/2018/10/17/opinion/brazil-election-fake-news-whatsapp.html
5. The saga of 'Pizzagate': The fake story that shows how conspiracy theories spread. [online]. Available: https://www.bbc.com/news/blogs-trending-38156985
6. False Rumor of Explosion at White House Causes Stocks to Briefly Plunge; AP Confirms Its Twitter Feed Was Hacked. [online]. Available: https://www.cnbc.com/id/100646197
7. Navigating the ‘infodemic’: how people in six countries access and rate news and information about coronavirus. [online]. Available: https://reutersinstitute.politics.ox.ac.uk/infodemic-how-people-six-countries-access-and-rate-news-and-information-about-coronavirus
8. Some Americans Are Tragically Still Drinking Bleach As A Coronavirus ‘Cure. [online]. Available: https://www.forbes.com/sites/nicholasreimann/2020/08/24/some-americans-are-tragically-still-drinking-bleach-as-a-coronavirus-cure/?sh=51804ab66748
9. It’s been exactly one year since Trump suggested injecting bleach. We’ve never been the same. [Online]. Available: https://www.politico.com/news/2021/04/23/trump-bleach-one-year-484399
10. van der Linden S, Roozenbeek J and Compton J (2020) Inoculating Against Fake News About COVID-19. Front. Psychol. 11:566790. doi: 10.3389/fpsyg.2020.566790.
11. Zubiaga, A., Liakata, M., & Procter, R. (2017). Exploiting context for rumour detection in social media. In Social Informatics: 9th International Conference, SocInfo 2017, Oxford, UK, September 13-15, 2017, Proceedings, Part I 9 (pp. 109-123). Springer International Publishing.
12. Wu, L., Morstatter, F., Hu, X., & Liu, H. (2016). Mining misinformation in social media. In Big data in complex and social networks (pp. 135-162). Chapman and Hall/CRC
13. Kaliyar, R. K., Goswami, A., Narang, P., & Sinha, S. (2020). FNDNet–a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, *61*, 32-44.
14. 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.
15. Ahmed, H., Traore, I., & Saad, S. (2017, October). Detection of online fake news using n-gram analysis and machine learning techniques. In *International conference on*

intelligent, secure, and dependable systems in distributed and cloud environments (pp. 127-138). Springer, Cham.

16. Yang, Y., Zheng, L., Zhang, J., Cui, Q., Li, Z., & Yu, P. S. (2018). TI-CNN: Convolutional neural networks for fake news detection. *arXiv preprint arXiv:1806.00749*.
17. O'Brien, N., Latessa, S., Evangelopoulos, G., & Boix, X. (2018). The language of fake news: Opening the black-box of deep learning-based detectors.
18. Singh, V., Dasgupta, R., Sonagra, D., Raman, K., & Ghosh, I. (2017). 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)* (pp. 1-3).
19. Purevdagva, C., Zhao, R., Huang, P. C., & Mahoney, W. (2020, December). A machine-learning based framework for detection of fake political speech. In 2020 IEEE 14th International Conference on Big Data Science and Engineering (BigDataSE) (pp. 80-87). IEEE.
20. Zhou, X., Jain, A., Phoha, V. V., & Zafarani, R. (2019). Fake news early detection: an interdisciplinary study. arXiv preprint arXiv:1904.11679.
21. Reddy, H., Raj, N., Gala, M., & Basava, A. (2020). Text-mining-based fake news detection using ensemble methods. International Journal of Automation and Computing, 17(2), 210-221.
22. Pérez-Rosas, V., Kleinberg, B., Lefevre, A., & Mihalcea, R. (2017). Automatic detection of fake news. *arXiv preprint arXiv:1708.07104*.
23. Gravanis, G., Vakali, A., Diamantaras, K., & Karadais, P. (2019). Behind the cues: A benchmarking study for fake news detection. *Expert Systems with Applications*, *128*, 201-213.
24. Shu, K., Wang, S., & Liu, H. (2018, April). Understanding user profiles on social media for fake news detection. In *2018 IEEE conference on multimedia information processing and retrieval (MIPR)* (pp. 430-435). IEEE.
25. Shu, K., Bernard, H. R., & Liu, H. (2019). Studying fake news via network analysis: detection and mitigation. *Emerging research challenges and opportunities in computational social network analysis and mining*, 43-65.
26. Choudhary, A., & Arora, A. (2021). Linguistic feature based learning model for fake news detection and classification. *Expert Systems with Applications*, *169*, 114171.
27. Mouratidis, D., Nikiforos, M. N., & Kermanidis, K. L. (2021). Deep Learning for Fake News Detection in a Pairwise Textual Input Schema, Computation 2021, 9, 20.
28. Wang, W. Y. (2017). " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*.
29. 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).
30. Zhou, Y. (2017). Clickbait detection in tweets using self-attentive network. *arXiv preprint arXiv:1710.05364*.

31. 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.
32. 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.
33. Qazi, M., Khan, M. U., & Ali, M. (2020, January). Detection of fake news using transformer model. In *2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)* (pp. 1-6). IEEE.
34. Singhania, S., Fernandez, N., & Rao, S. (2017). 3han: A deep neural network for fake news detection. In *Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Proceedings, Part II 24* (pp. 572-581). Springer International Publishing.