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 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

在当前的数字时代，假新闻的传播已成为政府、媒体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。本研究的主要目标是开发一种使用监督机器学习模型检测假新闻的高精度模型。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。 TF-IDF 向量化方法用于特征提取。该模型在一个大型数据集上进行训练和评估，该数据集是从 Kaggle 收集的，即虚假和真实的新闻数据集。该模型使用五种监督机器学习算法进行训练和测试：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。决策树分类器优于这些方法，达到了 99.57% 的惊人准确率。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。这些结果表明，与早期的最新结果相比，在检测虚假新闻方面取得了相当大的进步，证实了所建议的机器学习技术的有效性。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 the current digital era, 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 severe consequences for individuals and society. To address this issue, Machine learning techniques have emerged as a promising approach for detecting fake news. The primary goal of this study is to develop a model with high accuracy for detecting fake news using supervised machine learning models. The research methodology consists of three key steps: text pre-processing, feature extraction, and fake news detection. The TF-IDF vectorization method is used for feature extraction. The model is trained and evaluated on a large dataset, which is collected from Kaggle namely fake and real news datasets. The model is trained and tested using five supervised machine learning algorithms: Decision Tree Classifier, Logistic Regression, Passive Aggressive Classifier, Gradient Boosting Classifier, and Random Forest Classifier. The Decision tree classifier outperforms these methods, attaining a remarkable 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 results show considerable gains in detecting false news over earlier state-of-the-art results, confirming the efficacy of the suggested machine-learning technique. 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 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 do so instantaneously. However, with this vast ocean of information comes much noise. It can be challenging to discern what is fact from what is fiction and what is legitimate news from what is fake. Remarkably, this is true in social media, where anyone can post anything, regardless of its veracity. Social media has become so important that people use it more often than traditional news sources to get their news. According to a survey, 67% of Americans 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. Avaaz, a nonprofit organization, conducted a study found that more than 150 million people viewed political fake news in 2019. As absurd as those claims may appear, they were among the most often circulated false news articles on Facebook in 2019. Furthermore, a study conducted by Soroush Vosoughi, Ph.D., a computer scientist at Dartmouth University, and his coworkers discovered that misleading information spreads faster and reaches more people than the truth [2]. The report by Avaaz and the study by Dartmouth University highlight the concerning influence that bogus news can have on society. The spread of false information can lead to confusion, mistrust of legitimate sources, and even harm individuals and communities. Fake news and propaganda have become pervasive, and they can profoundly impact public opinion and discourse.

For a very long time, there has been fake news, and it was there before social media sites were developed. Fake news refers to misinformation, disinformation, or hoaxes distributed across traditional or social networking sites. It is a growing concern in today's society, as it may have detrimental effects, for example swaying people’s opinions, political influence, and even instigating violence. Fake news detection is identifying and verifying the accuracy of news articles and other content. It has gotten harder to distinguish between real and fake news as fake news has proliferated on social media and the internet and has become more sophisticated.

Detecting fake news 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 efficient in combating misinformation and disinformation. The phenomenon of spreading false news is not new. however as social media and the internet have grown, its impact has grown in recent years. The phrase "fake news" drew a lot of attention during the 2016 United States presidential election, when it was widely claimed that fake news was used to sway public opinion [3]. Similar ploys were utilized during the 2018 Brazilian elections when memes and heavily edited photographs were used to disseminate misinformation campaigns throughout that election cycle. A recent study found that 88% of the most widely shared images during that time leading up to the election of Brazil were either false or deceptive [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: According to this fake news report, Hillary Clinton and other prominent Democrats were involved in a child sex trafficking organization that used a pizza business in Washington, D.C. as a front. After the article gained popularity on social media, a man started shooting inside the eatery. The incident caused significant damage to the restaurant's reputation and resulted in a decline in sales [5].

(a) Twitter fake news (b) Dow Impact

Figure 1.1 Twitter fake news and its effect 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 repercussions can be far-reaching since it can erode public confidence in authorities, cause panic, and even inspire 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

In addition to a health disaster, the COVID-19 pandemic has also caused a significant infodemic with misinformation spreading rapidly through social media platforms and other online channels. False information 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 primary methods that people spread fake news during the COVID-19 situation is through social media. Platforms for social media like Facebook, Twitter, and Instagram have developed into a fertile ground for false information regarding 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 agenda, including promoting unproven treatments, conspiracy theories, or political propaganda. Reuters Institute for the Study of Journalism research revealed that false and misleading information about COVID-19 was more often disseminated on social media than factual facts, with a higher likelihood of engagement and sharing [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 causes significant problems in our society. One of the main problems was that it lead to confusion and fear among the public. False information about the transmission, symptoms, and treatment of the virus causes 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 caused vaccine reluctance, making it more difficult to build herd immunity and stop the virus's spread. 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, undermines public health measures, erodes public trust, and contributes to stigmatization and discrimination.

1.1.2 Machine learning

Machine learning is a field that falls under the umbrella of artificial intelligence and computer science. Its main objective is to develop algorithms that can learn and improve their accuracy over time, by analyzing data. In essence, machine learning aims to replicate the way humans learn and make decisions, with the help of computers [35].

- Supervised Learning
  Supervised learning is a sort of machine learning in which algorithms are trained on labeled datasets to properly categorize or predict data. During the training process, the model modifies its weights until it properly matches the data. To verify that the model does neither overfit nor underfit the data, cross-validation is utilized. Companies may use supervised learning to address a variety of practical issues on a wide scale, such as distinguishing spam emails from legal ones in our inboxes [35].

- Unsupervised Learning
  Unsupervised learning is a subset of machine learning in which algorithms are used to examine and cluster unlabeled samples. Without any human intervention, the algorithms may detect hidden patterns or groups in data. Unsupervised learning is very effective for exploratory data analysis, consumer segmentation, cross-selling techniques, picture and pattern recognition, and so on. It may also be used to minimize the number of features in a model through the application of dimensionality reduction techniques [35].

- Semi-Supervised Learning
  Semi-supervised learning is a subset of machine learning that combines supervised and unsupervised learning. During training, it uses a smaller labeled dataset to guide the classification and feature extraction process from a larger, unlabeled dataset. This method is especially beneficial when there is insufficient labeled data for a supervised learning algorithm or when labeling data is too costly. It is feasible to increase the algorithm's accuracy while minimizing the quantity of labeled data necessary by employing semi-supervised learning [35].

- Reinforcement Learning
  Reinforcement learning is a sort of machine learning that trains an algorithm to make decisions based on input from its surroundings. By pursuing behaviors that result in positive outcomes and avoiding actions that result in bad consequences, reinforcement learning aims to optimize a reward signal. Applications like playing games or operating robots, where the best course of action might not always be known in advance, benefit greatly from reinforcement learning. Through trial and error, the algorithm gains knowledge from its own experiences and gradually modifies its behavior to maximize the reward signal [35].

In conclusion, Machine learning is a powerful tool for tackling difficult problems in a range of industries, including banking, healthcare, and transportation. Machine learning algorithms are classified into four types: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. Each has benefits and drawbacks. Machine learning may be used to handle a wide range of real-world problems by distinguishing between these groupings and selecting the optimum solution for the task at hand.

1.2 Motivation for 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. To effectively combat false news, it is also imperative to address the shortage of varied and complete labeled datasets and incorporate user-friendly interfaces for communicating detection results.

By tackling these issues, the research hopes to aid in the creation of dependable and scalable solutions that lessen the propagation of false information, advance information integrity, and nurture a society that is more educated and resilient.

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 real 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?

The study seeks to advance our knowledge of the efficacy of machine learning techniques for fake news detection, offer insights into the difficulties and limitations of current approaches, and propose new techniques and strategies to improve the precision and usability of fake news detection systems.

1.5 Research Objective

To develop and evaluate a machine learning model when examining online news items for the presence of fake news, to increase the effectiveness and accuracy of fake news detection. We want to build and test a machine learning model for detecting fake news in online news articles that has a processing time of less than 10 seconds per article, an accuracy rate of at least 90%, and a false positive rate of less than 5%. A dataset of at least 40,000 news items from Kaggle will be used in the project, and the model will be assessed using cross-validation and a different test dataset.

1.6 Research Scope

- The study will concentrate on identifying fake news in English-language internet news stories.
- Based on many characteristics, including the article's source, language, and substance, the study will employ machine learning algorithms to classify news as real or fake.

- With an equal amount of true and false news pieces, the study will compile a dataset of at least 40,000 news articles.
- The machine learning model's performance will be assessed in the study utilizing cross-validation and a different test dataset.
- The development and evaluation of a machine learning-based fake news detection model will be the primary focus of the project, not its causes or effects.
- Other media types, such as social media or video content, will not be covered by the study.
- The study won't talk about the moral or legal ramifications of identifying or using fake news.

1.7 Thesis Organization

In the first chapter of our research paper, we introduce the topic and provide a brief background on the problem we are addressing. We then move on to Chapter 2, where we conduct a thorough literature review. This review includes an in-depth analysis of existing research and studies related to our topic, which helps us identify the gaps in the current knowledge and the research questions that we aim to answer.

Chapter 3 of our paper outlines the methodology we followed in our study. This section includes details on the datasets we used, the data preprocessing techniques applied, and a detailed explanation of our machine learning model. In Chapter 4, we present the experiments we carried out and the results obtained. We provide a detailed analysis of the data and the performance of our model. This section also includes visual representations of the results to help readers understand the findings better.

Chapter 5 is dedicated to the discussion and conclusion of our research. Here, we interpret our results in the context of the research questions we aimed to answer. We also discuss the limitations of our study and its implications for future research.

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.

The researchers (Yang et al., 2018) employed CNN (Convolutional Neural Network) to identify bogus news in their investigation. They used sensitivity analysis in their method and got a 92.10 percent accuracy rate. In their work, (O'Brien et al., 2018) classified bogus news using deep learning models. They used a DNN (black-box technique) in their study and got an accuracy of 93.50 percent. (Singh et al., 2017) looked into the application of various machine learning approaches for spotting fake news (Linguistic Analysis and Word Count). They were able to get an accuracy of 87.00 percent using the SVM (support vector machine).

(Purevdagva et al., 2020) developed a computerized system to identify false political speech. The system utilizes different methods of classification to gather information from political speeches and accompanying data, such as the topic of the speech, its location, the profile and credibility of the speaker, and contextual information. According to (Kaliyar et al., 2020), for pre-trained word embedding experiments, they substitute the processing layer's parameters with pre-trained word embedding vectors while retaining the indices and freezing the layer to avoid this from changing during the gradient descent procedure. Using FNDNet, they were able to obtain 98.12 percent.

(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.The importance of deep learning models over machine learning models for spotting fake news is the subject of several studies (Choudhary et al., 2021; Mouratidis et al., 2021). A hybrid convolutional neural network outperforms machine learning techniques in identifying fake news based on linguistic patterns, according to research by authors (Wang et al., 2017). With LSTM, (Rashkin et al., 2017) also had positive outcomes. Models focusing on linguistic features outperformed machine learning baselines. The proposed bi-directional gated recurrent unit comes from (Zhou et al., 2017). The Clickbait Challenge was won by the self-attentive system. Convolutional and recurrent neural networks combined in hybrid deep learning models that classify false news performed marginally better than non-hybrid baseline techniques (Nasir et al., 2021).

Modern pre-trained language models with remarkable performance across a range of natural language processing tasks, including the detection of fake news, include BERT and ALBERT. When compared to earlier state-of-the-art models, BERT improved the F-score on identifying fake news by 0.14, according to (Jwa et al., 2019). BERT, ALBERT, and comparable models have demonstrated outstanding performance in classifying bogus news, frequently outperforming conventional machine learning techniques. These models could aid in the creation of trustworthy and dependable methods for identifying and thwarting fake news in a variety of settings, such as news organizations and social media platforms. (Qazi et al., 2020) performs 15% more accurately than a hybrid CNN approach. Authors (Aggarwal et al., 2020) offer a work where they used a fine-tuned BERT model for fake news classification and attained an accuracy of 97.02 percent.

It has taken a lot of effort to categorize lengthy texts, like articles, as true or untrue. (Singhania et al., 2017) employed 3HAN, a deep learning technique that entails the creation of a vector for each article as well as a three-level bottom-up hierarchical article representation for words, phrases, and headlines. The Neural Network, which was trained on the notion that an article's headline should solely explain whether it is fake or not, receives this vector as input.

This chapter covered the related research on fake news detection that is currently accessible. The challenges posed by the linguistic features of news material compelled me to devise a solution. New computer science technologies, such as word embeddings and deep learning, provide me with the tools I need to solve the issue successfully. I will discuss the methodology and offer some theoretical insights into the approaches and algorithms I used to conduct this study in the next chapter.

CHAPTER 3

METHODOLOGY

3.1 Research design

The spread of false information is accelerating and increasing rate with the rise of communication and social media. Fake news detection is a new area of study that is gaining popularity [24]. The limited resources, including datasets, processing, and analytic methodologies, pose certain challenges. In this work, I develop a machine-learning model for detecting fake news. As a feature extraction method, I used the term frequency-inverse document frequency (TF-IDF), and as a programming language, I used python. I worked on five different machine learning models to compare each other in terms of their accuracy and evaluation matrix. To train these models, I also download the Fake and Real News datasets from Kaggle. (https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset). Here is the methodology and general work procedure diagram.

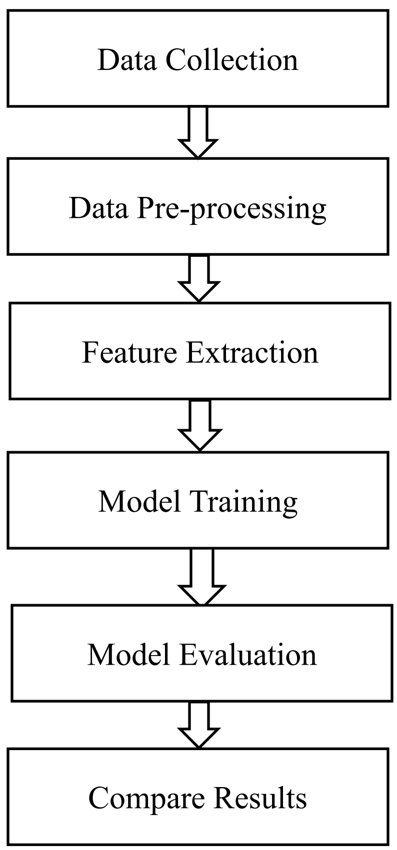

Figure 3.1 Work Procedure

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 to this dataset (https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset) and a brief description of the dataset:

"Fake.csv": This file contains 23,481 articles. The articles were collected from a variety of sources, including hoax websites, satire websites, and misleading news websites.

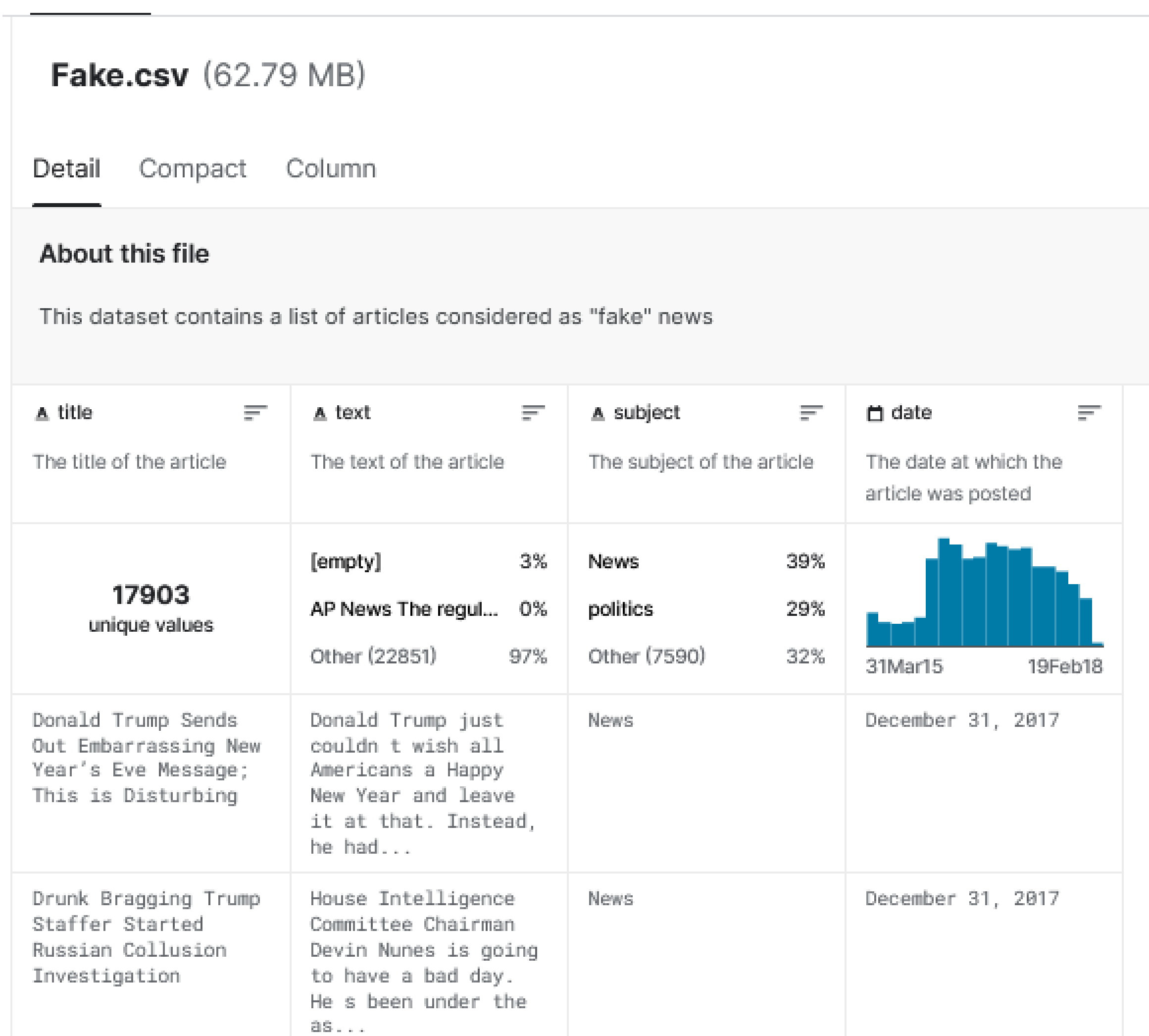

Figure 3.2.1 Fake News Datasets

"True.csv": This file contains 21,417 articles. The articles were collected from several sources, among them Reuters, the Associated Press, and other reputable news organizations.

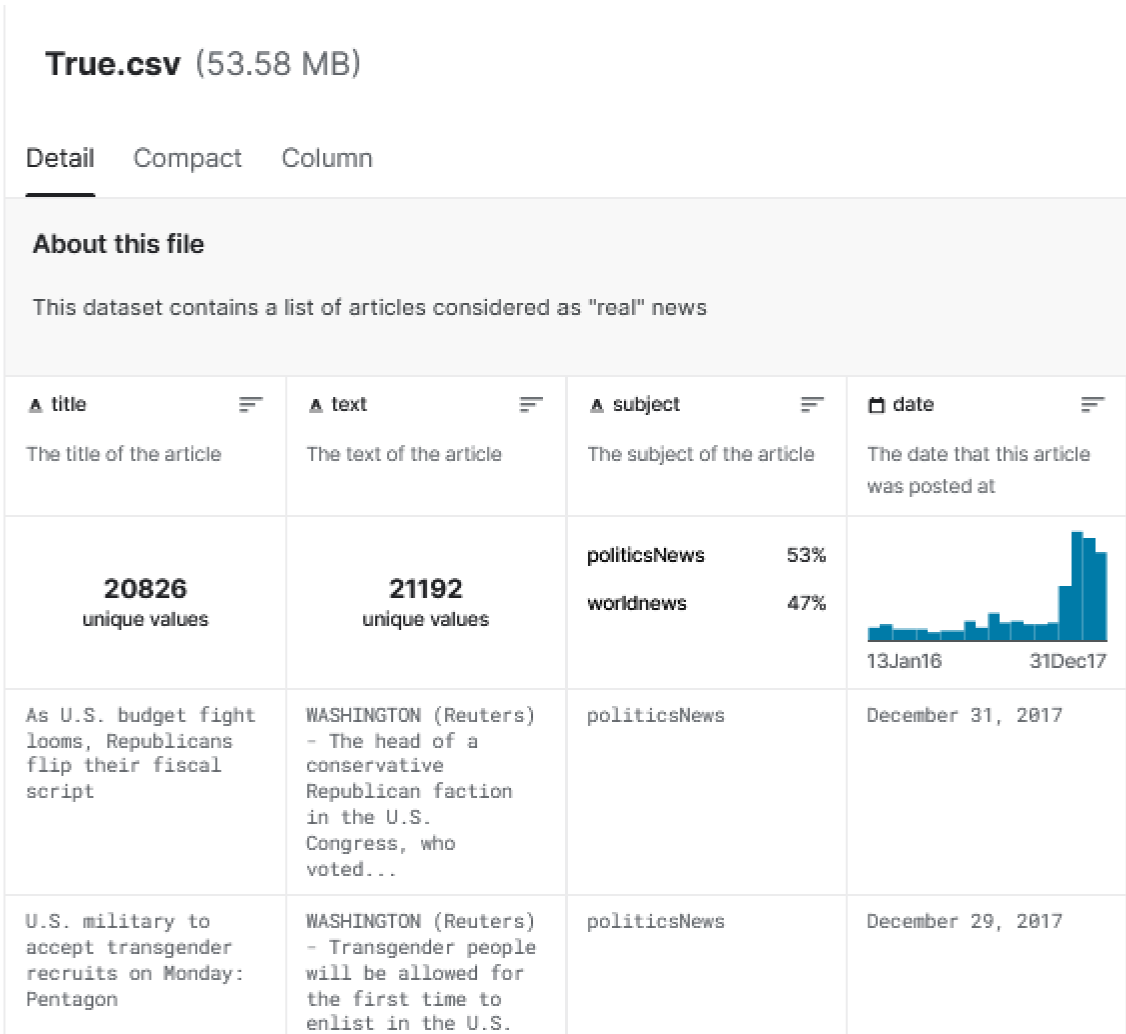

Figure 3.2.2 True News Datasets

The following columns are present in both files:

- "title": Article's title.
- "text": The main body of the article's text.
- "subject": Article’s subject.
- Date: Publication date of the article.

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.

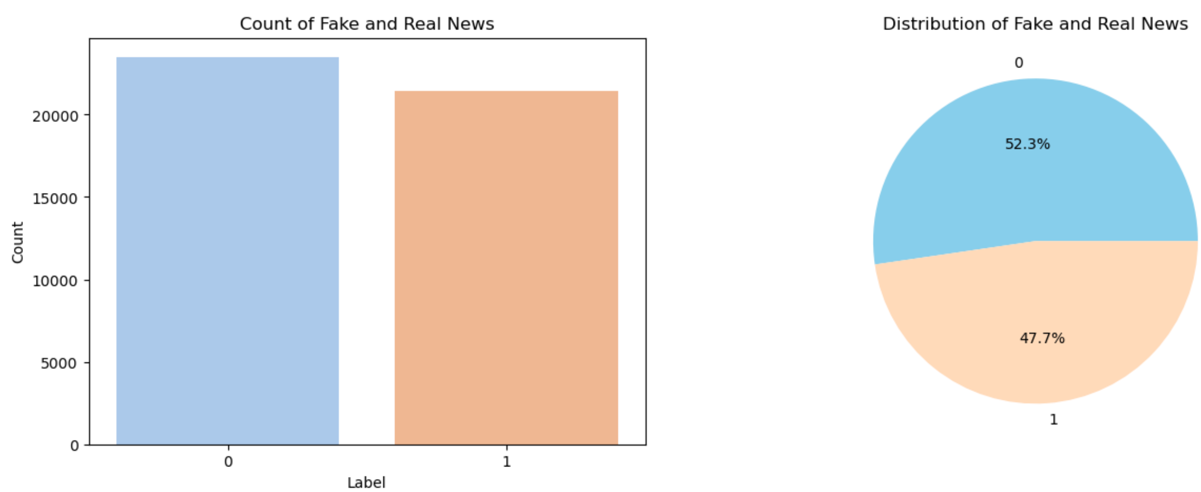

Figure 3.2.3 The number of fake and real news stories and their distribution.

In this dataset, we have a total of 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 kinds of news,

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

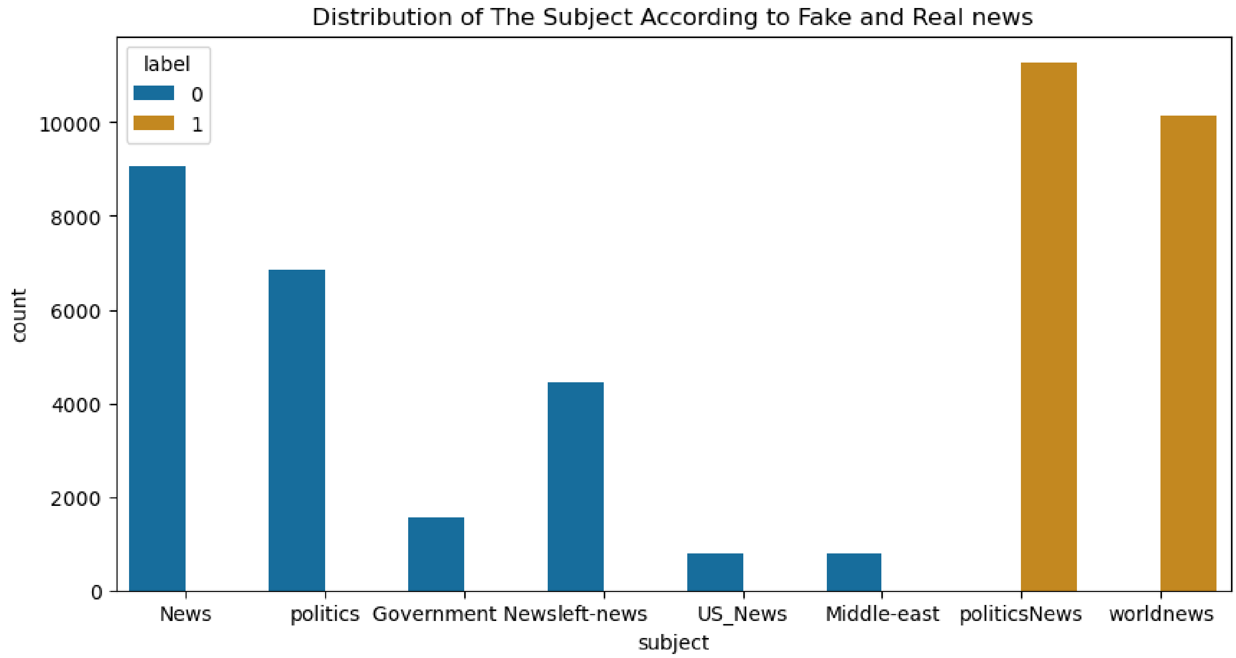

Figure 3.2.4 Distribution of The Subject According to Fake and Real news.

3.3 Data pre-processing

Data preparation is the procedure of altering, cleaning, and getting ready raw data for analysis. It is a crucial stage in the pipeline for data analysis cause the quality of the data used for analysis can significantly impact the accuracy and reliability of the results. Activities were completed to create the final dataset at this phase.

3.3.1 Data cleaning

After collecting the dataset, first, we need to clean the dataset. For cleaning the dataset, the steps involve handling missing values, outliers, and inconsistencies in the data. Depending on the data's structure and the analysis's conclusions, missing values may be re-added or eliminated. 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, my datasets were cleaned before so I didn’t need to clean dataset.

3.3.2 Data Integration

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

3.3.3 Data transformation

After that, a function is developed to pre-process the text data. Data often needs to be transformed to meet the assumptions of statistical tests for machine learning models to perform better. Common transformations include normalization (scaling the data to a specific range), log transformation (to handle skewed data), or power transformation (to stabilize variance).

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

- The news article's characters are all transformed into lowercase 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.

Term Frequency-Inverse Document Frequency, often known as TF-IDF [38], functions according to two main tenets. 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 considers a word's frequency over 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:

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

For each word in the corpus, the term frequency (TF) and inverse document frequency (IDF) are calculated as part of the TF-IDF vectorization procedure. 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. The TF-IDF score, which represents the word's importance within the text and the corpus as a whole, is created by multiplying these two values [38].

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. Several machine learning algorithms can be used for false news identification after the textual data has been translated into TF-IDF vectors. 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 use the TF-IDF numerical representations to determine if news stories are real or false.

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.

This research proposed a TF-IDF vectorization-based comprehensive approach for fake news detection. 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, increasing the accuracy of the detection of bogus news. 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. A potent method for automating the detection of fake news is provided by the combination of TF-IDF vectorization and machine learning algorithms, which helps to construct strong and dependable systems for battling disinformation.

3.3.5 Train test split

Train-test split is a frequent machine learning strategy for evaluating a model's performance on a dataset. The dataset is divided into two subsets: training and testing. The training set is used to train the model, while the testing set is used to assess how effectively the model generalizes to new, previously unknown 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 examine this function's parameters in more detail. The "X" variable typically represents the input features of the dataset, while "y" corresponds to the target variable or labels. We can effectively separate the data for training and testing purposes thanks to these variables.

The train size argument specifies how big the training dataset should be. 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 [39].

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.

Usually, the data is divided at random into a training set (which comprises roughly 70% to 80% of the data) and a test set (usually 20-30 percent of the data). In our scenario, we divided the data 80/20 between training and testing. After that, we used the training set to train our model and the test set to evaluate it. To gauge how well the model worked on fresh, untested data, the performance of the test set is used. Making sure the test set is reflective of the data as a whole is crucial. This indicates that the data distribution in the test set and the training set should be comparable. The model might not be able to generalize successfully to new data if the test set differs too much from the training set.

Other methods, such as cross-validation, which divides the data into many folds and uses each fold as a test set while training on the remaining folds, can be used in addition to a train-test split to assess a model's performance.

3.4 Machine learning models

After splitting the dataset into training and testing I use five different machine learning algorithms to find an algorithm that will fit in the model perfectly and accurately. Models with good f1 scores and confusion matrix can better perform. Let’s talk about this algorithm and how they work.

3.4.1 Decision Tree Classification

For both classification and regression tasks, a popular machine-learning technique known as Decision Tree Classification (DTC) is used. When classifying data into one or more classes, a tree-like model is built and a series of decisions are made depending on the attributes of the incoming data.

A training dataset is used to construct the decision tree, and each data point is represented by a set of characteristics and an associated class label. The procedure splits the data recursively into smaller subsets according to the characteristics' values. The division is carried out in a way that maximizes the gap between different classifications. A training dataset is used to construct the decision tree, and each data point is represented by a set of characteristics and an associated class label. The procedure splits the data recursively into smaller subsets according to the values of the characteristics. The division is carried out in a way that maximizes the gap between the various grades.

A choice is made based on the value of one of the features at each node of the tree. With this choice, the data is divided into two or more branches, each of which represents a different value for the characteristic. The procedure is repeated until only one kind of data is present at each leaf node. By following the decision path from the root node to a leaf node, a decision tree can be used to categorize fresh data points after it has been created. The algorithm moves to the appropriate child node depending on the value of each feature at each node and checks the value of the associated feature at each node [40]. The class label connected to that leaf node is returned as the anticipated class label for the input data point once the algorithm reaches a leaf node.

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 potent and popular machine learning method with applications in some industries, including marketing, finance, and healthcare.

3.4.2 Logistic Regression

Logistic regression (LR) is a popular machine-learning technique for classification applications. Logistic regression, as opposed to linear regression, which forecasts continuous values, forecasts the likelihood that an input will fall into a particular class.

In logistic regression, a set of characteristics serve as the input data, while a binary class label serves as the output (0 or 1). Modeling the likelihood of the input falling into the positive class (class 1) as a function of the input attributes is how the method operates. Any real-valued input is converted to a value between 0 and 1 using the logistic function [41] and it is known as such:

$$\mathrm{P} = \frac{1}{1 + e^{-z}}$$

where the input features are combined linearly to form z:

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

The logistic regression model's coefficients, b0, b1, b2,..., and bn, are here learned during training. The sigmoid function is another name for the logistic function.

The algorithm modifies the logistic regression model's coefficients during training to reduce the discrepancy between the anticipated probability and the actual class labels. Gradient descent or maximum likelihood estimates are frequently used for this. Following the training of the model, it may be used to classify new input data by computing the likelihood that the input corresponds to the correct class using the logistic function. If the probability is greater than a threshold, which is commonly 0.5, the input is classified as belonging to the positive class; otherwise, it is labeled as belonging to the negative class.

Due to several factors, logistic regression is a potent and often-used technique in machine learning. First off, it is a popular option for many applications because it is straightforward to understand. Second, it can manage input data that is both categorized and numerical. Third, it can be enhanced to manage multiclass classification jobs using methods like multinomial logistic regression or one-vs-rest.

Logistic regression, however, has several drawbacks. It presumes that the properties of the input and the output have a linear connection, which may not always be the case. Additionally, it presumes the independence of the input features, which may not hold in many applications in the actual world. Additionally, if the model is overly complicated or the input data contains a lot of characteristics, logistic regression may experience overfitting.

Overall, logistic regression is a potent and popular machine learning method with applications in a variety of fields, including marketing, finance, and healthcare.

3.4.3 Passive Aggressive Classifier

The Passive Aggressive Classifier (PAC) 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:

$$y = \sin(w.x)$$

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

During training, the PAC 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:

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

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 enables the technique to handle non-stationary situations where the underlying data distribution might change over time and swiftly adjust to changes in the incoming data [42].

Compared to other classification algorithms, the PA algorithm has several benefits. It can handle enormous datasets and is computationally efficient in the first place. Second, it can handle non-linearly separable data using kernel methods. Third, multiclass classification jobs can be handled with ease using one-vs-all or one-vs-one strategies.

The PA algorithm does, however, have significant drawbacks. If the learning rate is too high or the input data is noisy, overfitting may occur. Additionally, depending on the regularization value selected, it may need to be tuned for optimum performance. In general, the Passive Aggressive classifier is a robust and popular machine learning technique with applications in a variety of fields like text classification, picture classification, and anomaly detection.

3.4.4 Gradient Boosting

Gradient boosting, a popular machine-learning strategy, is used for both classification and regression issues. It is a particular kind of ensemble learning technique that combines numerous weak models, such as decision trees, to produce a powerful 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.

Gradient Boosting's main principle is to reduce the mean squared error or cross-entropy or loss function, which measures the discrepancy between the predicted values and the true values, using gradient descent. The approach calculates the gradient of the loss function concerning the forecasts of the prior model throughout each iteration [43]. The split points in a decision tree and other parameters of the new model are updated using this gradient. The sum of the predictions made by each model in the ensemble constitutes the ensemble's final forecast. This is the same as calculating a weighted average of the predictions, with the weights based on how well each model performed on the training set.

In comparison to other machine learning techniques, gradient boosting has several benefits. It can first handle both numerical and category data. Second, it can deal with input data that has missing values. Thirdly, it can identify non-linear connections between the properties of the input variable and the outcome variable. Gradient Boosting does, however, have some restrictions. If the model is overly complicated or the input data contains a lot of features, overfitting may also occur. Gradient Boosting may also need a lot of memory to hold the ensemble of models and can be computationally expensive.

Gradient Boosting is a robust and popular machine learning technique that has applications in a variety of industries, including marketing, finance, and healthcare. When high precision is necessary and there are intricate correlations between the features and the output variable in the input data, it is especially helpful.

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. A random subset of the data and characteristics are chosen for each tree during training, and the tree is then grown using a recursive partitioning technique. The method chooses the appropriate feature to split the data at each node of the tree based on some criterion, such as information gain or Gini impurity. Until the tree's maximum depth or number of nodes, the process is repeated [44].

Once every tree has been trained, it is joined to generate predictions based on fresh input data. The class label that is most commonly predicted by the individual forest trees makes up the final prediction. Comparing Random Forests to other machine learning methods, there are various benefits. It can first handle both numerical and category data. Second, it can deal with input data that has missing values. Thirdly, it can identify non-linear connections between the properties of the input variable and the outcome variable. There are limitations with Random Forests as well. How sensitive it is may depend on the choice of hyperparameters, including 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 expensive to compute, in particular.

Overall, Random Forests is a robust and popular machine learning method with applications in a variety of industries, including healthcare, finance, and marketing. When high precision is necessary and there are intricate correlations between the features and the output variable in the input data, it is especially helpful. Additionally, because Random Forests are capable of handling problems like noise and missing values, it is helpful when the input data is clean.

3.5 Model evaluation and comparison

Model evaluation and comparison are important steps in machine learning to assess the effectiveness of several machine learning techniques and select the ideal model for a specific issue. Several metrics and techniques can be used for model evaluation and comparison. Here are a few commonly used techniques:

Choosing evaluation metrics: Based on the nature of the problem, select the best assessment measures. For classification tasks, standard measurements include F1-score, accuracy, precision, recall, and area under the ROC curve (AUC-ROC). Metrics such as mean squared error (MSE), mean absolute error (MAE), and R-squared are commonly employed for regression concerns. Choose measurements that complement your challenge's unique aims.

Splitting the data: Datasets are typically partitioned into three parts: training, validation, and testing. The validation set is utilized to optimize hyperparameters and select models, the training set is utilized to train the model, and the test set is employed to assess the ultimate performance of the chosen model.

Accuracy: Accuracy is a frequent statistic for assessing classification algorithms. It calculates the percentage 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: When dealing with class imbalance, precision, and recall are two metrics used to evaluate the effectiveness of a classification algorithm. Recall measures the proportion of actual positive classifications among all true positive classifications, while precision measures the proportion of true positive classifications among all positive classifications. The f1 score, which is the harmonic mean of recall and precision, is a commonly used statistic for overall classification system evaluation.

Mean Squared Error (MSE): MSE (Mean Squared Error) is a common statistic for assessing regression algorithms. It computes the average of the squared discrepancies between expected and observed values. A drop in MSE suggests that the algorithm is doing better.

R-squared: R-squared is another statistic used to evaluate regression algorithms. It computes the proportion of the variance of the dependent variable that can be explained by the independent variables. The value of R-squared ranges from 0 to 1, with higher numbers signifying greater performance.

Cross-validation: Through the use of various subsets or folds of the dataset, a machine learning algorithm's performance is assessed using the cross-validation technique. On one subset of the data, the algorithm is taught, and on the other, it is assessed. Each subset is used as the test set once over this process's several iterations. The final performance statistic is calculated using the average performance across all folds. Cross-validation helps determine how well a model generalizes and for choosing hyperparameters that can be tweaked to enhance performance.

Training and tuning models: On the training set, use several techniques and hyperparameter combinations to train different models. Cross-validate the training set or utilize the validation set to modify hyperparameters and choose the best-performing model based on the evaluation metrics you've specified.

Evaluating models: Using the chosen evaluation criteria, assess the performance of each model on the test set. Compare the outcomes of various models to see which one performs the best. In machine learning, model evaluation and comparison are critical phases. The measurements and methodologies used for evaluation and comparison are determined by the type of problem and data. It is critical to use relevant measurements and approaches that offer a thorough review of the model's performance.

CHAPTER 4

EXPERIMENTS AND RESULTS

4.1 Experimental setup

To do this experiment I used a laptop with core i3 processor, 8GB ram, and 512 GB SSD. I also used Jupyter Notebook as a programming platform and Python as a programming language.

Jupyter Notebook: Jupyter Notebook is a web-based tool that enables users to generate and distribute documents containing code, equations, visualizations, and text. It can handle multiple programming languages and provides an interactive platform for machine learning, scientific computing, and data analysis. Due to its adaptability and user-friendly interface, it is extensively employed by educators, data scientists, and researchers.

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 [45]. Here are some important libraries I used:

- Numpy
  NumPy is a Python numerical computing toolkit that supports massive, multi-dimensional arrays and matrices, as well as a collection of mathematical functions for working with large arrays. It is well-known for its efficiency and ease of use in scientific computing, data analysis, and machine learning.

- Pandas
  Pandas is a Python data manipulation and analysis toolkit. It includes data structures for storing and handling massive datasets, as well as a variety of tools for data cleaning, transformation, and visualization.

- SciKit-learn
  Scikit-learn is a well-known Python machine-learning package that provides classification, regression, clustering, and dimensionality reduction techniques. It offers a variety of machine learning algorithms as well as tools for model selection, preprocessing, and evaluation. Scikit-learn is a Python library that is widely used in academic and industry contexts to design and deploy machine learning systems. It is noted for its versatility, ease of use, and interoperability with other Python libraries.

- Seaborn
  Seaborn is a Matplotlib-based Python data visualization package that provides a high-level interface for making useful and appealing statistical visuals.

- Matplotlib
  Matplotlib is a Python tool for creating static, animated, and interactive visualizations. It offers a variety of charts, plots, and graphs for representing data in various forms and is frequently used in scientific computing, data analysis, and machine learning. Matplotlib is well-known for its flexibility and ease of use, as well as its ability to build visually pleasing and useful data visualizations.

- 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

The process of evaluating the performance of a machine learning model and analyzing the output is referred to as results and analysis. The outcomes of a model's training and testing are examined to determine its correctness and efficacy in solving the problem at hand. For detecting fake news, I worked on five different Machine Learning models. Decision Tree Classifier (DTC), Logistic Regression (LR), Passive Aggressive Classifier (PAC), Gradient Boosting Classifier (GBC), and Random Forest Classifier (RFC) are the five models.

4.2.1 DecisionTreeClassifier (DTC)

The DTC 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.

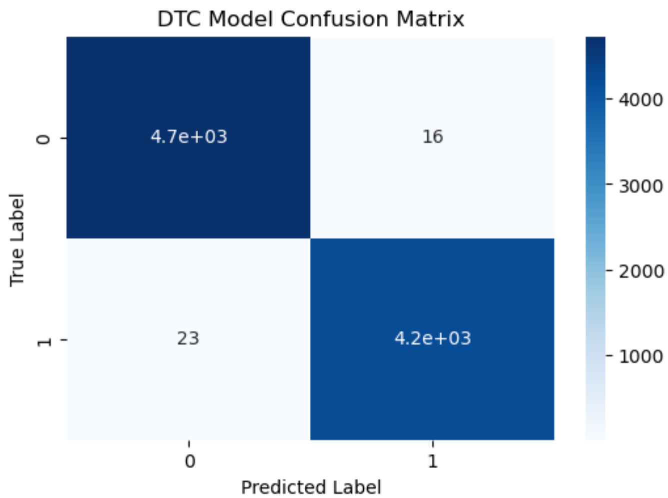

Figure 4.1 DTC Confusion Matrix

The confusion matrix shows how well the Decision Tree Classifier (DTC) model detects fake news. The matrix is a two-by-two table that displays the predicted true negatives (TN), false positives (FP), false negatives (FN) and true positives (TP).

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 reflect the amount of successfully predicted real and fake news items, respectively, while the false positives and false negatives show the number of misclassified articles.

Because the majority of predictions are true positives and true negatives, the matrix shows that the model has a high level of accuracy and precision. The low number of false positives and false negatives suggests that the model is effective at differentiating between real and fake news articles.

The confusion matrix indicates that the Decision Tree Classifier model is very accurate and dependable for detecting fake news, 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 LR 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.

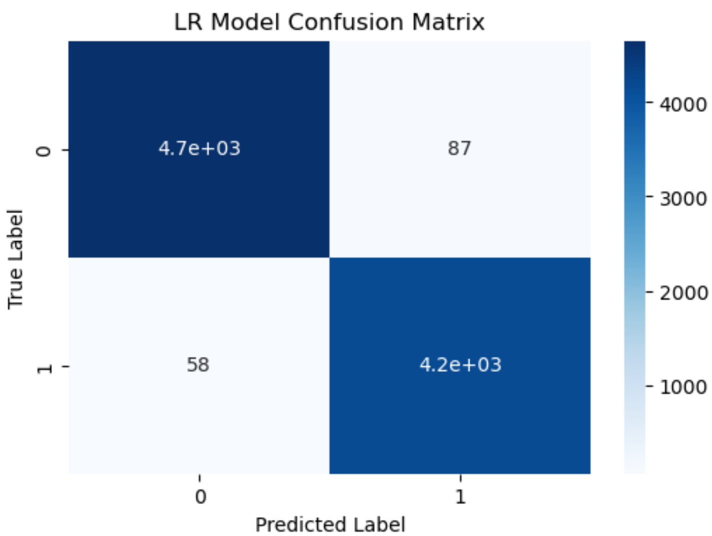

Figure 4.2 LR Confusion Matrix

The confusion matrix depicts the logistic regression model's performance in spotting bogus news. 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 reflect the amount of successfully predicted real and fake news items, respectively, while the false positives and false negatives show the number of misclassified articles.

The matrix demonstrates that the model has a high level of accuracy and precision, as the majority of predictions are true positives and true negatives. The number of false positives and false negatives, on the other hand, suggests that the model is not ideal and that there is room 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 presents the results of Logistic Regression, a machine-learning model for detecting fake news (LR). The precision score for class 0 (fake) is 0.99, while the precision score for class 1 (real) is 0.98. The recall score for class 0 (fake) is 0.98, while it is 0.99 for class 1 (real), showing that the model properly recognizes the majority of true and false news articles. The F1-score, which is the harmonic mean of accuracy and recall for both classes, is 0.98, showing that the model is equally capable of distinguishing between real and fake news.

4.2.3 PassiveAggressiveClassifier (PAC)

The PAC model scored an overall accuracy of 99.39 percent, indicating that it is quite good at guessing whether a news piece is authentic or phony.

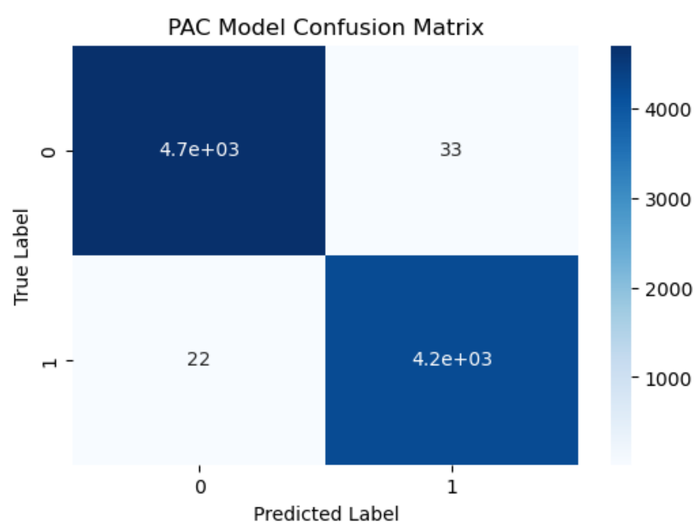

Figure 4.3 PAC Confusion Matrix

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 provides the findings of a machine learning model called Passive Aggressive Classifier (PAC) for detecting fake news. The precision score for class 0 (fake) is 1 and the recall score for class 1 (real) is 0.99, indicating that the model correctly detects the majority of real and false news articles. The F1-score, which is the harmonic mean of accuracy and recall, is similarly 0.99 for both classes, indicating that the model can discern between real and counterfeit news equally well.

4.2.4 GradientBoostingClassifier (GBC)

The GBC model attained an overall accuracy of 99.44 percent, indicating that it can correctly determine whether a news piece is authentic or phony.

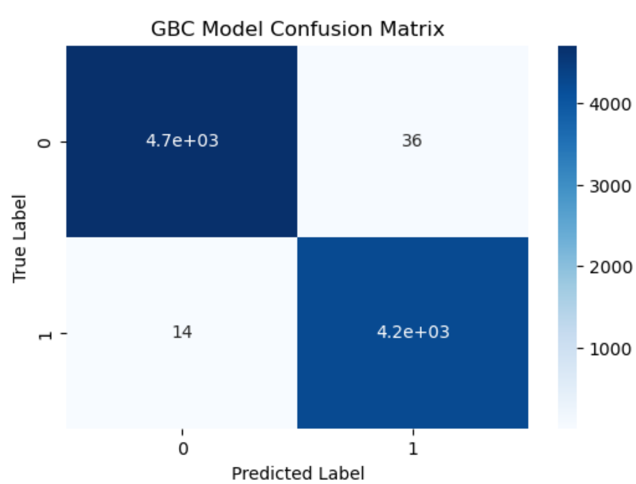

Figure 4.4 GBC Confusion Matrix

The confusion matrix shows how well the Gradient Boosting Classifier (GBC) model detects fake news. 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 reflect the amount of successfully predicted genuine and fake news items, respectively, while the false positives and false negatives show 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 provides the findings of a machine learning model called Gradient Boosting Classifier (GBC) for detecting bogus news. The accuracy and recall scores for both classes (genuine and fraudulent) are 1.00 and 0.99, respectively, indicating that the model correctly detects the majority of occurrences of true and phony news. The F1-score, which is the harmonic mean of accuracy and recall, is 0.99 for both genuine and false news, indicating that the model detects real news very well but detects fraudulent news somewhat less well.

4.2.5 RandomForestClassifier (RFC)

The algorithm attained an overall accuracy of 99.09 percent, indicating that it can correctly determine whether a news piece is authentic or phony.

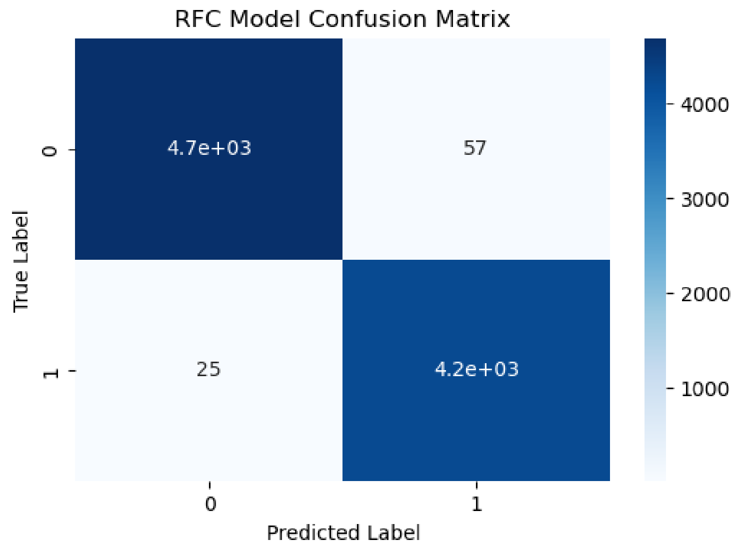

Figure 4.5 RFC Confusion Matrix

The confusion matrix illustrates the performance of the Random Forest Classifier (RFC) model in detecting fake news. The model accurately predicted 4682 true negatives and 4216 true positives in this confusion matrix, while mistakenly predicting 57 false positives and 25 false negatives. The number of accurately predicted actual and fake news pieces is represented by the true positives and true negatives, respectively, while the number of misclassified articles is represented by the false positives and false negatives.

The matrix demonstrates that the model has a high level of accuracy and precision, as the majority of predictions are true positives and true negatives. The number of false positives and false negatives, on the other hand, suggests that the model is not ideal and that there is room 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 displays the results of a Random Forest Classifier (RBC) machine learning model for detecting fake news. Both the precision and recall scores for both classes (genuine and fake) are 0.99, suggesting that the model accurately identifies the majority of actual and fake news events. The F1-score, which is the harmonic mean of precision and recall, is also 0.99 for both classes, showing that the model is equally capable of distinguishing between legitimate and fraudulent 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) has a maximum accuracy of 99.57 percent, which means it properly classifies the majority of the cases in the dataset. The precision score of 1.00 indicates that the DTC is usually always correct when predicting a positive class (F). Similarly, a recall score of 1.00 suggests that the DTC properly identifies almost all positive events. The F1-score of 0.99 suggests that precision and recall are balanced.

The Logistic Regression (LR) model outperforms the DTC model somewhat. It has a 98.39 percent accuracy, indicating that it successfully classifies a large amount of the dataset. The precision score of 0.99 indicates that the LR model correctly predicts a high proportion of the positive class. The recall score of 0.98 indicates that the LR model detects the majority of positive events but may miss a tiny percentage of them. The F1-score of 0.98 suggests a reasonable mix of precision and recall, but it performs somewhat worse than the DTC.

The Passive Aggressive Classifier (PAC) obtains 99.39 percent accuracy, indicating excellent overall performance. The precision score of 1.00 indicates that the PAC model rarely wrongly predicts positive cases. The recall score of 0.99 suggests that the PAC model successfully identifies the vast majority of positive events, but it may miss a few. The F1-score of 0.99 suggests that precision and recall are well balanced.

The accuracy of the Gradient Boosting Classifier (GBC) is 99.44 percent, which is quite close to the highest-performing model, DTC. The precision score of 1.00 implies that the GBC model correctly predicts a large proportion of the positive class. The recall score of 0.99 indicates that the GBC model successfully identifies the majority of positive events but may miss a few. The F1-score of 0.99 suggests that precision and recall are well balanced.

The Random Forest Classifier (RFC) obtains 99.09 percent accuracy, which is slightly lower than the top-performing models. The RFC model has a high proportion of true predictions for the positive class, as indicated by the precision score of 0.99. The recall score of 0.99 indicates that the RFC model successfully identifies the majority of positive events. The F1-score of 0.99 suggests that precision and recall are well balanced.

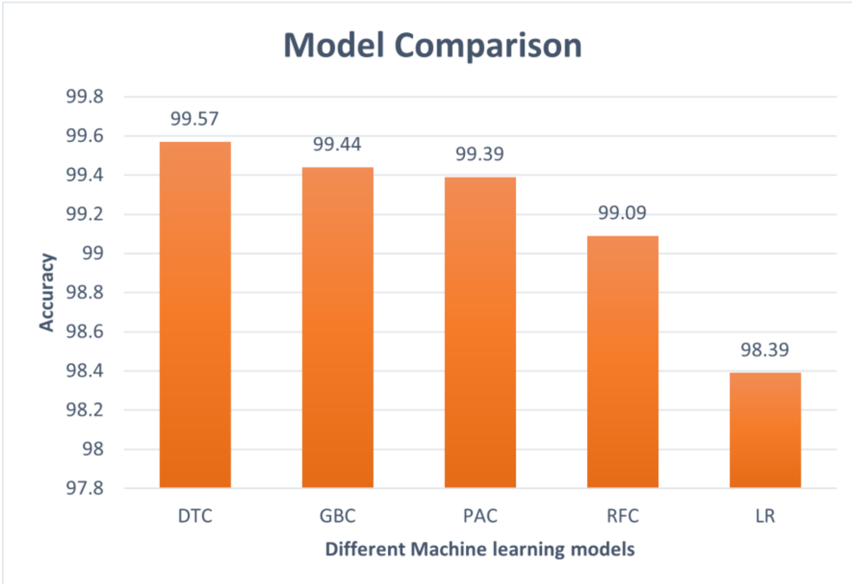

Figure 4.6 Model Comparison

Overall, all of the models perform admirably, with excellent accuracy and comparable precision, recall, and F1 scores. However, the accuracy data visualization shows that the DTC model performs very well, with the highest accuracy, while the LR model has the lowest accuracy. Finally, we may conclude that the Decision Tree Classifier is the most effective model.

CHAPTER 5

DISCUSSION

5.1 Error Analysis and model interpret

The confusion matrix shows the total number of true negatives (TN), false positives (FP), false negatives (FN), and true positives (TP). Based on the presented confusion matrices for multiple models (DTC, LR, PAC, GBC, RFC), the following analysis is performed:

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 Table

The DTC model has a large number of true negatives and true positives (correctly categorized negatives and positives), indicating that it is good at recognizing both groups.
It has a low number of false positives and false negatives, indicating a generally balanced and accurate model.

The LR model also has a high percentage of true negatives and true positives, indicating good classification performance. However, it has a larger number of false positives and false negatives than DTC, implying that it is more susceptible to misclassifications.

The PAC model performs similarly to DTC, with a high number of true negatives and true positives.
It has a low amount of false positives and false negatives, indicating strong classification accuracy.

In terms of true negatives and true positives, the GBC model surpasses DTC and PAC. It has somewhat more false positives than the prior models but far fewer false negatives.

The RFC model works well because it has a large number of true negatives and true positives.
It has a slightly higher rate of false positives but a relatively low number of false negatives when compared to DTC and PAC.

Based on the information we can say all of the models (DTC, LR, PAC, GBC, RFC) perform reasonably well, with high true negative and true positive counts. The occurrence of false positives and false negatives, on the other hand, highlights areas where the models might be improved. Further analysis, such as assessing misclassified cases, feature importance, or data distribution, can be undertaken to further understand the elements that contribute to these errors.

5.2 Implications and Mitigation.

In today's information ecosystem, where the quick spread of disinformation can have catastrophic effects, fake news identification and mitigation are crucial. Here are some consequences for detecting and mitigating fake news:

Advancements in Natural Language Processing (NLP): NLP techniques, such as machine learning and deep learning, are critical in detecting bogus news. 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. To determine the truth of statements, these systems can analyze enormous volumes of data and compare them to reputable sources. 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: Because of their massive user populations and algorithmic content distribution, social media platforms have a tremendous impact on the dissemination of fake news. 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 strategies should be used to ensure that these systems do not prejudice against specific perspectives mistakenly or increase 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 the promotion of 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 research

Fake news identification is a difficult and vital problem, but it also has significant limits and future development potential. Here are some potential constraints and future work opportunities:

Limited scope: Many existing fake news detection techniques are focused on detecting false news articles and may be incapable of detecting fake news propagated through other mediums such as social media posts, photos, or videos. Future work could focus on developing models that can detect fake news across a wider range of mediums and formats.

Data bias: The training data used to develop fake news detection methods may be biased. 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 detect fake 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 quickly on social media and other channels, necessitating real-time identification. 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 research could concentrate on building human-in-the-loop ways to improve the accuracy and reliability of false news identification by combining the benefits of machine-learning models with human knowledge and oversight.

Collaboration and benchmarking: Finally, enhancing the state of the art in false news identification will necessitate cross-disciplinary collaboration and benchmarking. 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.

CONCLUSION

In conclusion, the spread of fake news in this day and age is a serious threat to civilization, requiring effective detection mechanisms to preserve the integrity of information and promote informed decision-making. This study demonstrates the utility of machine learning techniques for detecting fake news articles on social 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 distinct supervised machine learning algorithms. The most successful model was the Decision Tree Classifier, attaining an accuracy rate of 99.57%. The research findings demonstrate the proposed technique's usefulness in outperforming previous state-of-the-art results and enhancing false news detection.

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. 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. More research and ingenuity are needed to address the persistent issue of fake news.

Future research could look into advanced machine learning algorithms, alternative feature extraction techniques, and the integration of multimodal data to improve the detection of fake news across many mediums. Additionally, the 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. Society can combat the threat of false news by utilizing the potential of machine learning to promote a more informed, accountable, and trustworthy information ecosystem for the benefit 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.
35. What is machine learning? [Online]. Available: https://www.ibm.com/topics/machine-learning
36. Bryanov, K., & Vziatysheva, V. (2021). Determinants of individuals' belief in fake news: A scoping review determinants of belief in fake news. *PLoS One*, *16*(6), e0253717.
37. Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, *19*(1), 22-36.
38. Peng, T., Liu, L., & Zuo, W. (2014). PU text classification enhanced by term frequency–inverse document frequency-improved weighting. *Concurrency and computation: practice and experience*, *26*(3), 728-741.
39. Tan, J., Yang, J., Wu, S., Chen, G., & Zhao, J. (2021). A critical look at the current train/test split in machine learning. *arXiv preprint arXiv:2106.04525*.
40. Priyam, A., Abhijeeta, G. R., Rathee, A., & Srivastava, S. (2013). Comparative analysis of decision tree classification algorithms. *International Journal of current engineering and technology*, *3*(2), 334-337.
41. LaValley, M. P. (2008). Logistic regression. *Circulation*, *117*(18), 2395-2399.
42. Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online passive aggressive algorithms.
43. Beygelzimer, A., Hazan, E., Kale, S., & Luo, H. (2015). Online gradient boosting. *Advances in neural information processing systems*, *28*.
44. Wainberg, M., Alipanahi, B., & Frey, B. J. (2016). Are random forests truly the best classifiers?. *The Journal of Machine Learning Research*, *17*(1), 3837-3841.
45. Zelle, J. M. (2004). *Python programming: an introduction to computer science*. Franklin, Beedle & Associates, Inc.