TU WIEN Informatics

Fake News Detection Performance Analysis by Incorporation of Sentiment Analysis

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

im Rahmen des Studiums

Wirtschaftsinformatik

eingereicht von

Parinaz Momeni Rouchi
Matrikelnummer 00828883

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao. Univ. Prof. Mag. Dr. Dieter Merkl

Signiert von: Parinaz Momeni Rouchi
Datum: 24.08.2023 14:11:55

A TRUST

Dieses Dokument ist digital signiert!
Dieses mit einer qualifizierten elektronischen Signatur versehene Dokument hat gemäß Art. 25 Abs. 2 der Verordnung (EU) Nr 910/2014 vom 23. Juli 2014 ("eIDAS-VO") die gleiche Rechtswirkung wie ein handschriftlich unterschriebenes Dokument.

Prüfinformation: Informationen zur Prüfung der elektronischen Signatur finden Sie unter: www.a-trust.at/pdf

Wien, 3. August 2023

Parinaz Momeni Rouchi

Dieter Merkl

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Informatics

Fake News Detection Performance Analysis by Incorporation of Sentiment Analysis

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieurin

in

Business Informatics

by

Parinaz Momeni Rouchi

Registration Number 00828883

to the Faculty of Informatics

at the TU Wien

Advisor: Ao. Univ. Prof. Mag. Dr. Dieter Merkl

Vienna, August 3, 2023

Parinaz Momeni Rouchi

Dieter Merkl

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Parinaz Momeni Rouchi

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Signiert von: Parinaz Momeni Rouchi
Datum: 24.08.2023 14:15:55

TRUST

Dieses Dokument ist digital signiert!
Dieses mit einer qualifizierten elektronischen Signatur versehene Dokument hat gemäß Art. 25 Abs. 2 der Verordnung (EU) Nr 910/2014 vom 23. Juli 2014 ("eIDAS-VO") die gleiche Rechtswirkung wie ein handschriftlich unterschriebenes Dokument.

Prüfinformation: Informationen zur Prüfung der elektronischen Signatur finden Sie unter: www.a-trust.at/pdf

Wien, 3. August 2023

Parinaz Momeni Rouchi

Danksagung

Zuerst möchte ich meinem Betreuer Prof. Dieter Merkl herzlich für seine unschätzbare Unterstützung und Anleitung während des gesamten Prozesses der Erstellung dieser Arbeit danken. Seine Expertise, hilfreiche Vorschläge und konstruktive Kritik haben maßgeblich zur Qualität dieser Arbeit beigetragen.

Ich möchte auch meine Wertschätzung gegenüber all meinen Professoren zum Ausdruck bringen, die mit ihrer Geduld, Zeit und Anleitung ein positives Lernumfeld geschaffen haben.

Abschließend möchte ich meinen tiefsten Dank an meine Eltern, Parvin Rajabzadian und Ali Momeni Rouchi, meine Schwester Dr. Elaheh Momeni-Ortner und meinen Verlobten Mag. Sebastian Grabner aussprechen, die mich während meines Studiums kontinuierlich unterstützt und ermutigt haben.

Acknowledgements

First of all, I would like to thank my supervisor Prof. Dieter Merkl, for his invaluable support and guidance throughout the entire process of preparing this thesis. His expertise, helpful suggestions, and constructive criticism have been instrumental in shaping the quality of this work.

I would also like to express my appreciation to all my professors who have created a positive learning environment with their patience, time, and guidance during each course.

Lastly, I would like to convey my deepest thanks to my parents, Parvin Rajabzadian and Ali Momeni Rouchi, my sister, Dr. Elaheh Momeni-Ortner, and my fiance, Mag. Sebastian Grabner, for their continued support and encouragement throughout my studies.

Kurzfassung

Online-Medienplattformen wie soziale Netzwerke und Online-Nachrichtenplattformen haben sich als Hauptquellen für den Zugriff auf Nachrichten etabliert, aufgrund ihrer geringen Kosten, einfachen Zugänglichkeit und Attraktivität. Infolgedessen verbringen Menschen zunehmend mehr Zeit auf diesen Plattformen. Die große Datenmenge und die Entwicklung von Fake News mithilfe raffinierter Deep-Learning-Algorithmen erschweren jedoch Experten die manuelle Überprüfung der Inhalte. Daher ist die Entwicklung automatisierter Fact-Checking-Tools und Lösungen zur Erkennung von Fake News unerlässlich, um die Verbreitung nicht überprüfter Informationen über verschiedene Plattformen hinweg einzudämmen. Die Vielzahl verfügbarer Ansätze erschwert jedoch die Auswahl der geeignetsten Methode für spezifische Anwendungsfälle von Forschern und Praktikern. Zudem variiert die Charakteristik von Fake News in unterschiedlichen Domänen. Das Hauptziel dieser Forschung ist daher die Bewertung bestehender Methoden und die Durchführung eines umfangreichen Leistungsvergleichs von Machine-Learning- und Deep-Learning-Algorithmen, einschließlich Support Vector Machines (SVM), Naive Bayes und einem Deep Neural Network, unter Verwendung verschiedener Datensätze aus verschiedenen Domänen. Die Studie untersucht auch den Einfluss der Einbeziehung von Sentiment-Analyse auf die Klassifikationsleistung, um Erkenntnisse über die Effektivität der Sentiment-Analyse als ergänzende Komponente zu gewinnen. Durch diese umfassende Bewertung soll die Arbeit den Entscheidungsprozess erleichtern und bei der Auswahl einer geeigneten Methode zur Fake News-Erkennung für individuelle Anwendungsfälle unterstützen.

Die Forschung beginnt mit einer umfassenden Literaturrecherche, um geeignete Algorithmen und vorherrschende Domänen für die Aufgabe zu identifizieren. Anschließend werden die ausgewählten Algorithmen auf vier verschiedene Datensätze angewendet und trainiert, die Politik, Gesundheit, Klimawandel und soziale Medien repräsentieren. Die Leistungsbewertung unter Verwendung von Testdaten aus jeder Kategorie zeigt, dass SVM konsistent bessere Ergebnisse erzielt als andere Algorithmen und die höchste Genauigkeit erreicht. Das neuronale Netzwerk zeigt jedoch bessere Leistung bei unausgewogenen Datensätzen, was auf sein Potenzial bei der Handhabung solcher Datenverteilungen hinweist. Es ist anzumerken, dass unausgewogene Datensätze negative Auswirkungen auf neuronale Netzwerke haben können, was zu Overfitting und geringerer Verallgemeinerungsfähigkeit für Minderheitsklassen führt. Dennoch erfordert das neuronale Netzwerk-Modell deutlich weniger Rechenaufwand. Darüber hinaus zeigen die Ergebnisse, dass die Einbeziehung

von Sentiment-Analyse keine signifikanten Verbesserungen bringt und in einigen Fällen sogar zu leicht schlechterer Leistung führt. Dies ist auf die unterschiedliche Verteilung der Sentiment-Klassen zwischen Fake News und echten Nachrichten innerhalb jedes Datensatzes zurückzuführen. Daher wird festgestellt, dass die alleinige Einbeziehung von Sentiment-Analyse als ergänzendes Merkmal die Gesamtleistung der Fake News-Erkennung nicht verbessert. Es wird jedoch empfohlen, eine Kombination aus einem neuronalen Netzwerk und einem ausgewogeneren Datensatz zu verwenden, um sowohl Ressourcenbeschränkungen als auch Leistungskennzahlen zu berücksichtigen.

Abstract

Online media platforms, such as social networks and online news platforms, have emerged as primary sources for accessing and consuming news, due to their low cost, ease of access, and attractiveness. Consequently, individuals are increasingly spending more time on these platforms. However, the vast volume of data and the evolution of fake news through the use of sophisticated deep learning algorithms present challenges for experts to manually examine the content. As a result, the development of automated fact-checking tools and fake news detection solutions has become essential to combat the propagation of unverified information across diverse platforms. However, the multitude of available approaches complicates the task of researchers and practitioners in selecting the most appropriate method for their specific use cases. Moreover, the characteristics of fake news varies across different domains. Therefore, the primary objective of this research is to assess the existing methods and to conduct a large-scale performance comparison of machine learning and deep learning algorithms, including Support Vector Machines (SVM), Naive Bayes, and a Deep Neural Network, utilizing diverse datasets from various domains. The study also explores the impact of incorporating sentiment analysis on the classification performance, aiming to provide insights into the effectiveness of sentiment analysis as a supplementary component. By undertaking this comprehensive evaluation, the thesis aims to facilitate the decision-making process and aid in the selection of an appropriate fake news detection method for individual use cases.

The research begins with a comprehensive literature review to identify appropriate algorithms and prevalent domains for the task. Subsequently, the selected algorithms are implemented and trained on four distinct datasets representing politics, health, climate change, and social media. Performance evaluation using test data from each category reveals that SVM consistently outperforms other algorithms, achieving the highest accuracy. However, the neural network demonstrates better performance when confronted with imbalanced datasets, highlighting its potential in handling such data distributions. It is noted that imbalanced datasets can negatively impact neural networks, leading to overfitting and reduced generalization for minority classes. Nonetheless, the neural network model requires significantly less computational effort. Furthermore, the findings indicate that the inclusion of sentiment analysis does not lead to significant improvements and, in some cases, even results in slightly lower performance. This can be attributed to the varying distribution of sentiment classes between fake news and real news within each dataset. Consequently, it is concluded that incorporating sentiment

analysis as a complementary feature alone does not enhance the overall performance of fake news detection. However, it is suggested that a combination of a Neural Network with a more balanced dataset achieves promising outcomes considering both resource constraints and performance metrics.

Contents

CHAPTER 1

Introduction

1.1 Motivation

1.1.1 Fake News

Fake news refers to information that is deliberately created to deceive its audience and can be presented in the form of text, image, video, or audio. There are three key aspects of fake news that are often highlighted: The presence of deceptive intent, the degree to which the content is verifiably false, and the use of different platforms such as online news websites, social media, or e-commerce websites (mainly targeting online product reviews) to publish false information [AVGRV21, CTZ20].

Fake news contributors can be categorized into three types, namely bots, trolls, and cyborg users [SSW+17]. Fake content can be created and spread by non-human accounts, and bots are accounts controlled by algorithms that can create a false impression that a piece of information is popular, leading to the spread of fake news [SCV+18]. For example, social bots focus on social media sites. However, fake accounts are not the only contributors; trolls are real humans who aim to disrupt a community by creating doubts in users' minds. A cyborg is a combination of fake and human contributors, where the accounts are created by real humans, but programs are used to perform certain tasks.

Misinformation, disinformation, and fake news are three terms that are commonly associated with false information, but the distinction between them lies in the intent behind the creation of false content [HHK+23].

- *Misinformation* – Inaccurate or false information that is spread unintentionally, often due to a lack of knowledge or understanding.
- *Disinformation* – Intentionally false information that is spread with the goal of deceiving or manipulating the audience.

- *Fake News* – Fake news is a broader term that encompasses both misinformation and disinformation and includes forged stories or hoaxes that are presented as if they were real news. They often mimic the form of mainstream news [SSW+17, ZM20].

Understanding the distinctions between these terms is important in designing effective approaches to detect and combat the spread of false information in the online environment.

1.1.2 Fake News Detection

The challenges related to the credibility of online news have played a significant role in shaping the practices of news verification. However, the abundance of large-scale data available on the Internet and online environments such as social media, micro-blogging sites, and news websites, as well as the continuous evolution of fake news, has made it challenging for experts to examine the contents manually. Consequently, the quality of news shared in online environments is often lower than the traditional news sources, and the content is less reliable. Misinformation and biased stories can have negative effects on individuals, society, corporations, and governments[DRP+18, SMW+20]. Examples of fake news related to the 2016 US presidential election and the COVID-19 pandemic have raised significant concerns worldwide and demonstrated that misinformation can spread in various domains, including health, politics, and finance. In recent years, there has been a growing number of studies that aim to automatically identify fake news, misinformation, or propaganda using AI and Natural Language Processing techniques.

The problem of fake news detection is complex and multidimensional, with various factors affecting it. As a result, different approaches have been categorized based on their focus. Recent studies have explored either individual or a combination of the following categories: [AVGRV21, SQJ+19]:

- *Knowledge-based, fact-checking approach* that aims to evaluate the reliability of news using semantic web and linked open data [WAL+14, CSR+15]. One of the commonly employed subcategories within this approach is the Knowledge Graph-based (KG) approach. KGs are graph-structured knowledge bases that integrate facts from various sources [MSS22].

- *Context-based approach* that aims to assess the truthfulness of a news article by analyzing its metadata, including user behavior, news source, and the community that interacts with it. Such an approach can reveal insights into the article's credibility or bias. This category considers following features:
  - Social features
    - User Network – Number of followers and followees
    - Post Information – Number of replies, likes or shares
  - Spatiotemporal Information

  - ∗ Spatial – Users or news from different locations
  - ∗ Temporal – News or responses from different Timestamps

- *Content-based approach* that endeavors to assess the validity of a story by analyzing its content. This category considers following features:

  - Linguistic – News articles with text
  - Visual – News articles with images

  In addition, various features can be extracted from the content such as sentiment.

Each approach can alleviate the challenges that other approaches encounter. Depending on the study's objective and the selected approach, one may opt for a single or a combination of machine learning techniques. Some researchers opt to conduct their fake news research using unsupervised methods as they claim that supervised methods require an extensive amount of time and labor to build a reliable annotated dataset. For instance, Gangireddy et al. [GLC20] developed a graph-based approach for fake news detection in the absence of labeled historical data. However, several available fake news datasets can overcome this challenge. As such, this work follows the content-based approach that utilizes sentiment as latent feature and proposes a benchmarking framework based on different supervised learning algorithms and annotated datasets from various domains.

1.1.3 Sentiment Analysis

Sentiments refer to the emotions or opinions expressed by individuals in relation to entities through text, images, or sounds. Sentiment analysis is a natural language processing discipline that automates sentiment classification through machine learning techniques. Document-level sentiment analysis analyzes the content of entire paragraphs or documents and provides an overall sentiment score, while sentence-level sentiment analysis assigns independent scores to each sentence in the document [VPS18].

Sentiment analysis classifiers are designed to categorize emotions into classes such as "positive", "negative", or "neutral", as depicted in Figure 1.1.

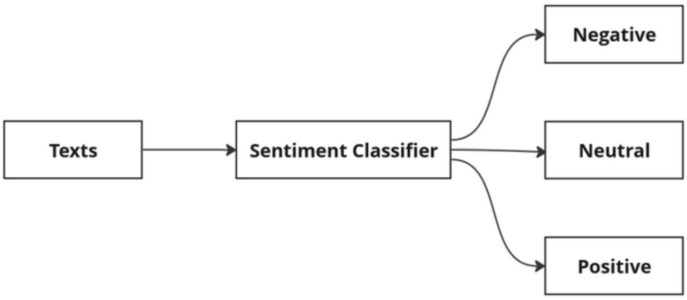


Figure 1.1: Sentiment Analysis

However, other binary classifications with only two classes, such as "positive" and "negative," also exist. Choosing the appropriate granularity level for sentiment analysis requires careful consideration of the size, context, polarity, and subjectivity of the text corpus. Binary classifiers face several challenges and may fail to classify some texts correctly, such as dual-polarity sentences like "The location was amazing ... but the food was not good." Moreover, in natural language processing, two types of text can be distinguished: subjective and objective. Subjective texts, such as "The car is good," express opinions or sentiments, while objective texts, such as "The car is red," do not contain explicit sentiments and can be classified as neutral. However, some cases can be challenging for even fine-grained sentiment classifiers. For instance, when sarcasm or irony are involved in a text, people may express their negative sentiments using positive words, which can be difficult for the algorithm to detect. Despite these challenges, a fine-grained sentiment analysis can provide more accurate results.

Text-based sentiment analysis algorithms can be broadly classified into two categories: machine learning-based algorithms and dictionary-based algorithms. The choice of algorithm depends on the language, scope, and availability of the corresponding dictionaries. Dictionary-based algorithms rely on pre-existing dictionaries of sentiment-laden words, whereas machine learning-based algorithms learn from the input data and do not require pre-existing dictionaries [HB16]. This thesis focuses on the task of three-way classification of sentiment, namely "positive", "negative", or "neutral", and employ machine learning-based algorithms to accomplish this task.

1.1.4 Sentiment Analysis for Fake News Detection

The sentiment derived from sentiment analysis can influence the spread of fake news. Depending on their motives, individuals may write positively or negatively about a subject. For example, on social media platforms, users may express negative sentiment when a trending topic does not benefit them or align with their personal interests [GS15]. The study by Dickerson et al. [DKS14] demonstrated the feasibility of differentiating human Twitter accounts from social bot accounts through the use of sentiment variables. The ROC curves presented in Figure 1.2 provide an evaluation of the performance of two classifiers: one that incorporates sentiment variables and another that does not. Through visual analysis, it becomes evident that the classifier utilizing sentiment variables demonstrates better performance, as indicated by its closer proximity to the true positive rate. The ROC plot[1] demonstrates the effectiveness of incorporating sentiment variables in improving the classification performance.

[1] A receiver operating characteristic curve plot, or ROC curve plot, provides a visual representation of the trade-off between the true positive rate and the false positive rate for a binary classification model, allowing for an assessment of its discriminative power and overall effectiveness.

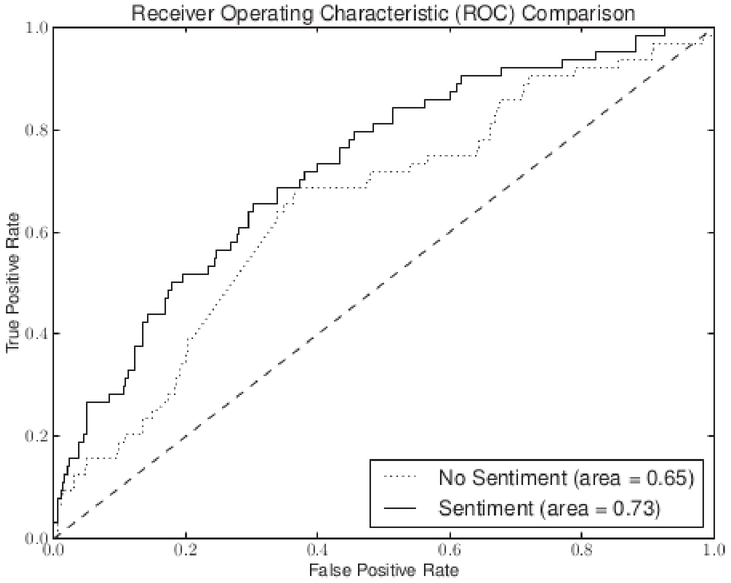


Figure 1.2: The receiver operating characteristic (ROC) curves compare the performance of the best classifier both with and without the inclusion of sentiment features. [DKS14].

It is not by chance that fake news publishers intentionally aim to increase the spread of fake news through clickbait[2], expressing strong emotions or polarity (positive or negative) to mislead the audience, as studies have shown that the majority of readers only read the headlines [CCR, HA17]. Therefore, text sentiment can be deemed an effective feature in the process of fake news detection. This thesis aims to compare the performance of various fake news detection classifiers by incorporating sentiment analysis as a complementary element.

[2] Clickbait refers to online content, such as headlines or thumbnails, designed to attract attention and encourage users to click on a link.

1.2 Problem Statement

The exponential growth of news on online platforms has rendered manual examinations unfeasible. Consequently, there has been a proliferation of automated solutions for detecting fake news. Moreover, the availability of increasingly powerful processors has led to the proliferation of fact-checking tools across various platforms. These tools are constantly evolving, with major hyperscalers such as Microsoft, Google and Amazon offering advanced solutions for fake news detection. Microsoft NewsGuard[3], Google Fact Check Tools[4] and Amazon Neptune ML[5] are exemplary solutions offered by these tech giants.

The high-performance pre-processing and training of complex Deep Learning models based on large amounts of unstructured text data, whether on cloud computing products or on-premise infrastructure, necessitates intense computational efforts. To mitigate the complexities associated with these computations, Del Ser, Javier, et al. [DSBL$^+$] proposed an innovative approach that leverages a bagging ensemble technique based on randomization of recurrent neural networks. This technique has proven to be highly effective in mitigating computational complexities in the pre-processing and training stages of deep learning methods.

Automated fake news detection tools, whether deployed on cloud services or on-premise infrastructure, can be useful, but their potential is constrained in various ways. For example, content-based approaches, which rely on natural language processing techniques, tend to perform better on certain language formats, for instance, even in advance English, which is the most widely used language in existing solutions, they can fail when attempting to detect false news in regional languages. Moreover, automated solutions are less effective in verifying new and unchecked claims, such as those that may be based on a limited dictionary of words. This limitation is particularly evident in Google Fact Check Tool. Furthermore, platform type, whether social media or online news, context, text size (e.g., news articles have longer text length than social media posts), and cultural and environmental factors can all affect the accuracy and performance of fake news detection methods. These factors can lead to different results and performances, making it challenging for data scientists and researchers to identify the most suitable set of criteria to consider in their solutions, and accurately estimate their performance and implementation effort in the rapidly evolving and heterogeneous solution landscape.

As previously discussed in Section 1.1.4, the creators of fake news employ various tactics to achieve their objectives, including the use of emotional appeals to engage their audience. This highlights the importance of incorporating sentiment analysis in the classification process of fake news. While some reviews have highlighted the potential benefits of including sentiment as an additional feature, there are relatively few studies that have empirically investigated the effectiveness of this approach in practice.

[3] https://www.newsguardtech.com/ (*Last access as of August 3, 2023*)

[4] https://toolbox.google.com/factcheck/explorer (*Last access as of August 3, 2023*)

[5] https://aws.amazon.com/neptune/machine-learning/ (*Last access as of August 3, 2023*)

To summarize, there is a limited number of published works that have compared multiple fake news detection key factors, while also considering latent features. This work aims to address this gap by comparing different supervised learning algorithms and datasets from various domains, and by incorporating sentiment analysis as a complementary approach to achieve a comprehensive performance comparison.

1.3 Goal and Research Questions

The objective of this thesis is to provide a performance comparison of various algorithms using diverse datasets, enabling researchers and practitioners to preliminarily evaluate and select the most suitable algorithm for their individual needs and use cases. The performance evaluation includes models trained both with and without sentiment analysis as a complementary feature. This study aims to provide comprehensive insights and specific recommendations to facilitate the selection and implementation of a suitable fake news detection solution for a given application.

To effectively address the identified problem, the following research questions were formulated.

1.3.1 Research Question 1 (RQ1)

Which techniques exist for fake news detection and how can they be categorized?

1.3.2 Research Question 2 (RQ2)

How can the incorporation of sentiment analysis enhance the effectiveness of fake news detection models?

1.3.3 Research Question 3 (RQ3)

How can the proposed solution be implemented through a comparative case study using multiple test datasets?

1.3.4 Research Question 4 (RQ4)

What conclusions can be drawn from a performance comparison of the implemented solution?

1.4 Methodology and Approach

This work utilizes the techniques of literature reviews and multiple case studies.

1.4.1 Literature Review

The purpose of conducting a comprehensive literature review is to critically examine and analyze existing research publications on fake news detection. This review serves to gain insights into the various techniques and their underlying mechanisms employed in this field. Moreover, it helps identify key domains relevant to fake news detection, aiding in the selection of appropriate test datasets and performance metrics for the subsequent multiple case study and performance comparison.

The literature review in this work employs a systematic search strategy that involves defining appropriate search terms to filter the relevant scientific publications. In the initial step, these search terms are applied in conjunction with the search functions of prominent publication databases, such as the Association for Computing Machinery (ACM), the Institute of Electrical and Electronics Engineers (IEEE), and ScienceDirect. The used search terms and the number of results are summarized in Table 1.1, grouped according to research questions. The subsequent step involves identifying the 20 most recent publications within the last 10 years, specifically those published since 2013, that meet the established criteria for scientific research. These publications are presented as relevant results in Table 1.1, thereby ensuring the credibility and validity of the study. The techniques employed within these selected publications are then documented for thorough examination and analysis.

Research Question	Search Terms	Database	Results	Relevant Results
Which techniques exist for fake news detection and how can they be categorized?	Fake News Detection Techniques	IEEE	246	15
		ACM	541803	
		ScienceDirect	2538	
	Supervised Fake News Detection	IEEE	77	
		ACM	310287	
		ScienceDirect	1376	
	Content-Based Fake News Detection	IEEE	12	
		ACM	629677	
		ScienceDirect	1889	
How does sentiment analysis improve fake news detection models?	Sentiment Analysis for Fake News Detection	IEEE	84	10
		ACM	543448	
		ScienceDirect	781	
	Sentiment Analysis for Fact Checking	IEEE	17	
		ACM	564878	
		ScienceDirect	15733	

Table 1.1: Search strategy of the literature review

1.4.2 Case Study

In the practical phase of this study, a hybrid approach is adopted, integrating elements of both exploratory and explanatory case study designs, in alignment with the findings of the literature review. Initially, an exploratory approach is employed to gain deeper insights into the selected datasets after incorporating sentiment as an additional feature. For the purpose of conducting a comprehensive comparative case study, a Full-Factorial design is proposed. This design involves implementing various classification algorithms to address the challenge of fake news detection, using multiple test datasets from different domains. In the explanatory part of the study the performance of these algorithms is investigated under two conditions: with the inclusion of sentiment as an additional linguistic feature, and without the incorporation of sentiment. By systematically exploring these variations, the study aims to provide a comprehensive analysis and comparison of the effectiveness of different classification approaches for fake news detection.

By employing a combination of multiple data sources and diverse analytical methodologies, this work incorporates both data source diversity and methodological diversity, ensuring a robust examination of the research problem from different perspectives. Additionally, the criteria for case selection, as outlined by Verner et al. [VST^{+}09] in their work on defining guidelines for multiple case studies, are presented in Table 1.2.

Case Criterion	**Addressing in Research Design**
The case should be precise and unambiguous.	This criterion is ensured through the literature review that encompasses relevant approaches of fake news detection. Clear categories of cases are derived to facilitate the selection of appropriate algorithms and test datasets.
All information about processing and evaluation of the case should be accessible to the researcher.	Access to the relevant test datasets, machine learning libraries and performance metrics is addressed in the implementation phase of this study.
The case should be conducive to achieving the research goals.	The evaluation of each individual case is crucial to fulfill the overarching research goal of conducting a comprehensive full-factorial experiment.

Table 1.2: Criteria for case selection.

Following the implementation phase, a performance evaluation is carried out by assessing the performance metrics identified during the literature review. Subsequently, a comparison of all solutions takes place, adhering to the guidelines for multiple case studies defined by Verner et al. [VST^{+}09] (as listed in Table 1.3).

Criterion for performance metrics for case evaluation	**Addressing in Research Design**
The performance metric is precise and clear.	The utilization of mathematically computed performance indicators, along with their corresponding calculation formulas and units, is employed.
The performance metric is relevant to the concepts being measured.	Conducting a comprehensive literature review on the topic of performance evaluation of fake news detection algorithms is an essential component of the theoretical phase of this study.

Table 1.3: Criteria for case selection.

By incorporating these criteria for case selection and performance metrics, the study ensures a clear selection of cases for analysis but also enhances the validity and reliability of the research findings.

1.5 Structure of the Work

Chapter 2 encompasses a comprehensive literature review aimed at addressing RQ1 and RQ2. In the pursuit of answering RQ1, a comprehensive collection of prevalent techniques employed in fake news detection is conducted to be able to derive requirements for the subsequent selection and construction of test datasets to be used in the implementation and evaluation phases. Subsequently, research question RQ2 is addressed through the identification of potential applications of sentiment analysis in the context of fake news detection. By examining existing literature, this investigation aims to provide an understanding of the role and impact of sentiment analysis in improving the accuracy and reliability of fake news detection outcomes. In addition, the performance evaluation approaches used fake news detection solutions in prior studies will be analyzed for later implementation. As a culmination of the literature review, an overall comparison is followed in the form of a catalog of solutions.

To address RQ3, the findings from the literature review will be used as requirements and general conditions for implementing the solutions within a comparative case study presented in Chapter 3. Diverse test datasets sourced from different application domains are utilized for this purpose. Following the implementation phase, a performance evaluation is conducted, enabling a comparison of the various solutions using the predefined performance metrics. Following the completion of implementation and performance evaluation, Chapter 4 undertakes an analysis and categorization of the obtained results.

Moreover, the generalizability of specific phenomena observed throughout the implementation and evaluation phases will be discussed in more detail in Chapter 5. Additionally, Chapter 5 addresses the limitations of the study, which impact the achieved outcomes.

Further, Chapter 6 encompasses an overall comparison of results, providing final conclusion. By highlighting these overall assessments and taking into account the implications and limitations discussed in Chapter 5, this thesis aims to provide a substantiated answer to RQ4.

Finally, Chapter 7 explores possible areas for future research, based on the findings and insights obtained in the study.

CHAPTER 2

Literature Review

2.1 Which techniques exist for fake news detection and how can they be categorized?

As outlined in Subsection 1.1.2, the majority of studies on fake news predominantly adopt a content-based approach. Within the domain of content-based fake news research, various techniques are employed, including supervised, unsupervised, and Deep Learning methods [CTZ20]. Supervised techniques rely on labeled data and involve partitioning the dataset into training and testing sets. The training set is utilized for model training, while the test set is used for prediction evaluation. Prominent researchers such as Buntain et al. [BG17], Shu et al. [SMW$^+$20], and Castillo et al. [CMP13] have employed this technique successfully to develop machine learning models for text classification in the context of fake news detection. Notable examples of traditional machine learning supervised methods for fake news detection include: Naive Bayes Classifier, Support Vector Machines (SVM), Random Forests, Logistic Regression, Decision Trees and Gradient Boosting Methods (e.g., XGBoost, AdaBoost). These methods have been widely used in the field of fake news detection and have shown promising results in different research studies. Unsupervised methods have as well gained significant traction in the detection of deceptive content. This approach leverages unlabeled data as input, eliminating the need for partitioning the data into training and testing sets. Unlike supervised techniques, unsupervised methods do not rely on predefined labels for training, but instead focus on extracting patterns, clusters, or anomalies within the data. Some researchers aim to exclusively conduct their research on fake news utilizing unsupervised methods, arguing that supervised approaches necessitate a substantial investment of time and effort to construct a dependable annotated dataset (Yang et al. [YSW$^+$19]). In line with this perspective, Gangireddy et al. [GLC20] devised a graph-based approach for detecting fake news in scenarios where labeled historical data is unavailable. Some of the well-known traditional machine learning unsupervised methods for fake news detection include:

- Clustering algorithms: Techniques such as k-means clustering, hierarchical clustering, and DBSCAN have been applied to group similar news articles based on their content and identify patterns or anomalies indicative of fake news.

- Topic modeling: Methods like Latent Dirichlet Allocation (LDA) and Non-Negative Matrix Factorization (NMF) have been utilized to extract latent topics from a collection of news articles. By analyzing the distribution of topics within articles, patterns related to fake news can be uncovered.

- Network analysis: Approaches leveraging network analysis, such as community detection algorithms and centrality measures, can reveal the structure and connections among news sources or social media users. Identifying suspicious patterns or influential sources can assist in detecting fake news propagation. However, it should be noted that the network analysis approach falls under the category of context-based approaches, as discussed in Subsection 1.1.2.

Another line of research primarily emphasizes the utilization of deep learning models, such as recurrent neural networks (RNN) or convolutional neural networks (CNN) [ARVB16, VSVR16]. Despite their typical applications in image processing and audio sources, both CNN and RNN have demonstrated outstanding performance in the field of natural language processing (NLP). RNNs have been employed in various submissions [Baj17, ZZX17] for the *fake news challenge*[1] competition. Collobert et al. [CWB$^+$11] leveraged a CNN model to acquire generic representations across various NLP benchmarks, such as Part-Of-Speech (POS) tagging and semantic role labeling.

2.1.1 Conclusion on the Fake News Detection Techniques

With the help of the relevant publications, three categories of the most commonly used approaches have been identified that are widely employed in the field. The two first categories encompass traditional supervised machine learning methods and the second category focuses on deep learning techniques, which have gained significant attention due to their ability to extract intricate patterns and representations from complex data. A summary of the identified categories, along with relevant works within each category, is provided in Tables 2.1, 2.2 and 2.3.

[1] http://www.fakenewschallenge.org/ (*Last access as of August 3, 2023*)

Category	Relevant Work	Methodology & Performance
Linear Models	Detecting Fake News Using Support Vector Machines[2]	This paper proposes a SVM model that classifies news articles with 89% accuracy based only on title, and a 98% accuracy with the title and news article.
	Truth and deception at the rhetorical structure level [RL15]	The SVM classifier is used with RST and VSM to classify news as true or fraudulent with an accuracy of 86%.
	Detecting opinion spams and fake news using text classification [ATS18]	This paper developed a SVM model for credibility analysis with TF-IDF and reached an accuracy of 83%.

Table 2.1: Identified category and relevant works – Linear Models

[2] `https://mehtaplustutoring-mlbootcamp20.github.io/Real_vs_Fake_News/` (*Last access as of August 3, 2023*)

Category	Relevant Work	Methodology & Performance
Probabilistic Models	A Benchmark Study on Machine Learning Methods for Fake News Detection [KKI+19]	This study employed various traditional machine learning methods and yielded an accuracy of 95% with Naive-Bayes (with n-gram).
	A multistage credibility analysis model for microblogs [AAQAR+15]	In this work a credibility analysis is conducted to identify fake content with Naive-Bayes which achieving an accuracy 90.3% using Twitter data.
	A smart system for fake news detection using machine learning [JSKG19]	In this paper the researchers use both probabilistic and linear classifiers, Naive-Bayes and SVM respectively, and achieve a better performance upto 93.6% accuracy with SVM model.

Table 2.2: Identified category and relevant works – Probabilistic Models

Category	Relevant Work	Methodology & Performance
Deep Neural Network Models	Neural Stance Detectors for Fake News Challenge [ZZX17]	A multi-perspective matching neural net architecture is proposed for both headline and body text of to analyze Fake News Challenge (FNC1) dataset which is further able to reach metric score close to 87%.
	Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks [LW18]	In this work, both RNN and CNN-based methods are used to build propagation paths for detecting fake news at the early stage of its propagation.

Table 2.3: Identified category and relevant works – Deep Neural Network Models

The three identified categories have emerged as the most popular methods for fake news detection due to their effectiveness and versatility. These approaches offer robust probabilistic modeling capabilities, allowing for the assessment of the likelihood of news articles being fake or genuine. Linear models, on the other hand, provide a straightforward and interpretable framework for feature analysis and prediction. Additionally, neural networks, with their ability to capture complex patterns and relationships in data, have as well demonstrated promising performance fake news detection tasks. The widespread adoption and effectiveness of these methods have established them as the go-to choices for researchers in the field.

2.2 How can the incorporation of sentiment analysis enhance the effectiveness of fake news detection models?

Saif M. Mohammad, in his book "Sentiment Analysis: Detecting Valence, Emotions, and Other Affectual States from Text" [Moh16], emphasizes the role of sentiment analysis in annotating tweets to identify user intents, including criticism, support, ridicule, and more. By leveraging sentiment analysis, researchers can gain insights into the emotional tones and subjective attitudes expressed in text, enabling a deeper understanding of authors' intentions related to a piece of news or users' intentions related to a social media post. Sharma et al. [SQJ+19] explored computational techniques for detecting and mitigating fake news and observed that positive sentiment words were often exaggerated in positive fake reviews compared to genuine ones, while responses to fake news on social media tended to exhibit a negative sentiment. For example, users' comments such as "Do not spread lies!" or "Stop hoax!" serve as indications of fake news propagation. Thus, the integration of sentiment analysis using machine learning techniques allows for the evaluation of agreement and disagreement within social media posts and news articles. This facilitates the differentiation between factual news and misleading information.

Lin et al. [LKL22] proposed a novel approach based on Bidirectional Encoder Representation from Transformers (BERT) for analyzing the relationship between text sentiment and the presence of harmful news. Their study focused on the categorization of information disorder types, as outlined by Wardle et al. [WD17], as depicted in Figure 2.1.

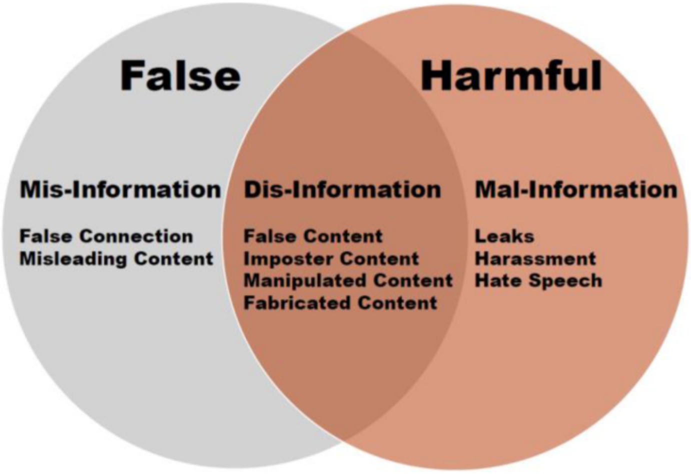


Figure 2.1: Information Disorder Diagram [WD17].

By utilizing the power of BERT, Lin et al. aimed to uncover the correlation between sentiment and various forms of misinformation, thereby contributing to a deeper understanding of the impact of sentiment in the context of harmful news detection, while also elucidating its relevance to the broader domain of fake news. In addition, they showed the relevance of harmful news to fake news.

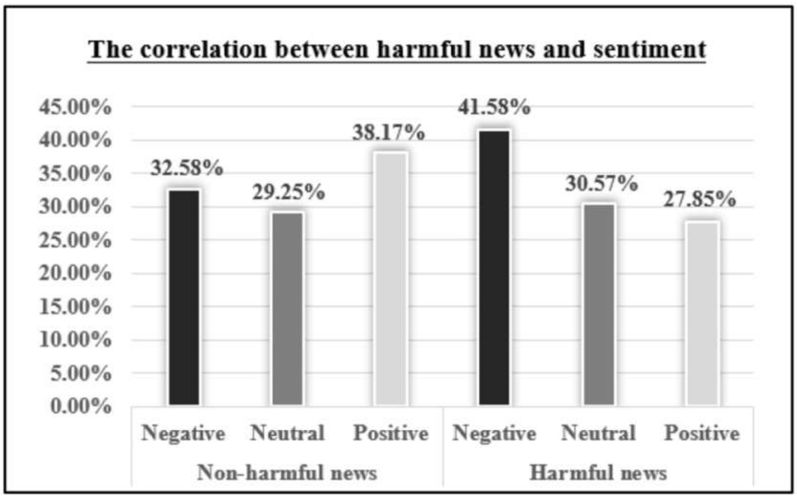


Figure 2.2: Correlation between harmful news and sentiment [LKL22].

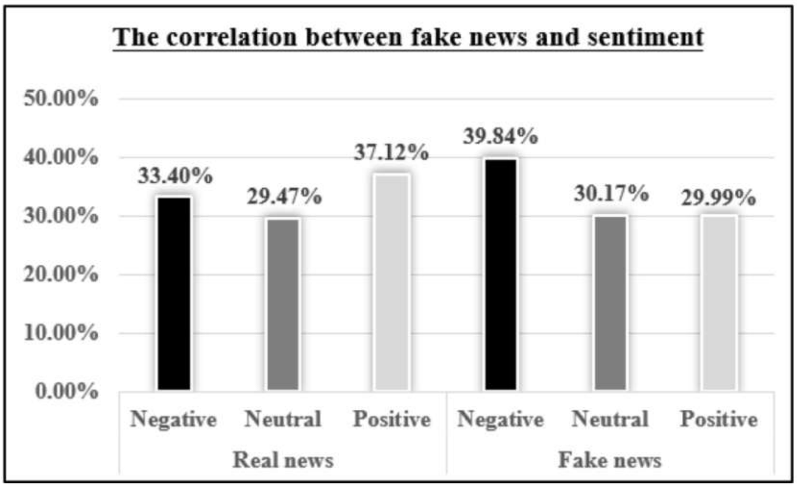


Figure 2.3: Correlation between fake news and sentiment [LKL22].

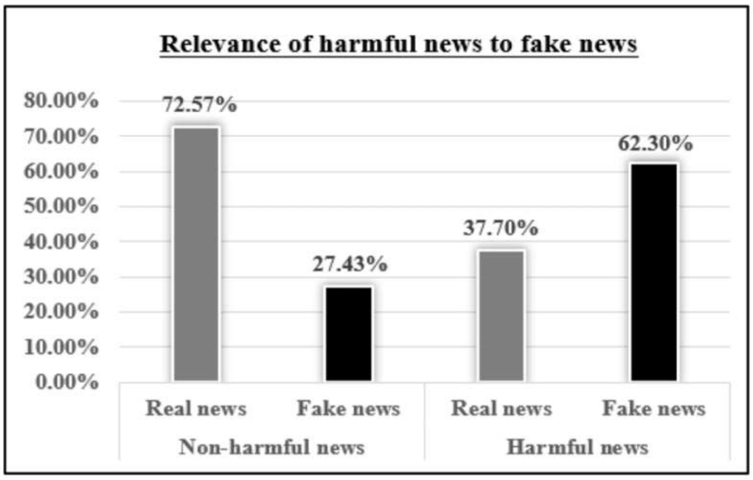


Figure 2.4: Relevance of harmful news to fake news [LKL22].

The analysis of Figure 2.2 reveals a notable pattern concerning the relationship between harmful news and sentiment proportions. Harmful news exhibits a distinctively higher proportion of negative sentiments, while non-harmful news tends to have a higher proportion of positive sentiments. This observation holds true for the comparison between fake news and real news as well, as Figure 2.3 depicts. Figure 2.4 shows that within the category of non-harmful news, approximately 72.57% of the news content corresponds to real news, while among the category of harmful news, around 62.3% of the news content can be attributed to fake news. Although the correlation between harmful news and fake news is not particularly strong, a positive correlation can still be observed. This highlights the significance of considering sentiment as a distinguishing factor for various types of news content.

2.2.1 Using Sentiment as a Complementary Feature in Fake News Detection

One of the primary steps in constructing effective fake news datasets is defining the features that contribute to building a robust prediction model, as highlighted by Castillo et al. in their work [CMP13]. Working with a comprehensive features allows researchers to explore various approaches. Among the crucial features identified by Castillo et al. [CMP13], 3 out of 10 best-performing features were sentiment related ones: sentiment score, number of positive sentiment words, and number of negative sentiment words.

Furthermore, as briefly discussed in Section 1.4, Dickerson et al. [DKS14] introduced an ensemble learning system in their research, which incorporated a range of sentiment-based features, as well as users' profile and network features, to distinguish between human users and bots on Twitter. Their study demonstrated that the sentiment-aware classifier outperformed the classifier without sentiment features. They conducted a comparison of the established features to determine the ones that had the greatest impact on the best-performing classifier. Figure 2.5 illustrates the 25 most important features they identified, revealing that 19 of the top 25 features are sentiment-related.

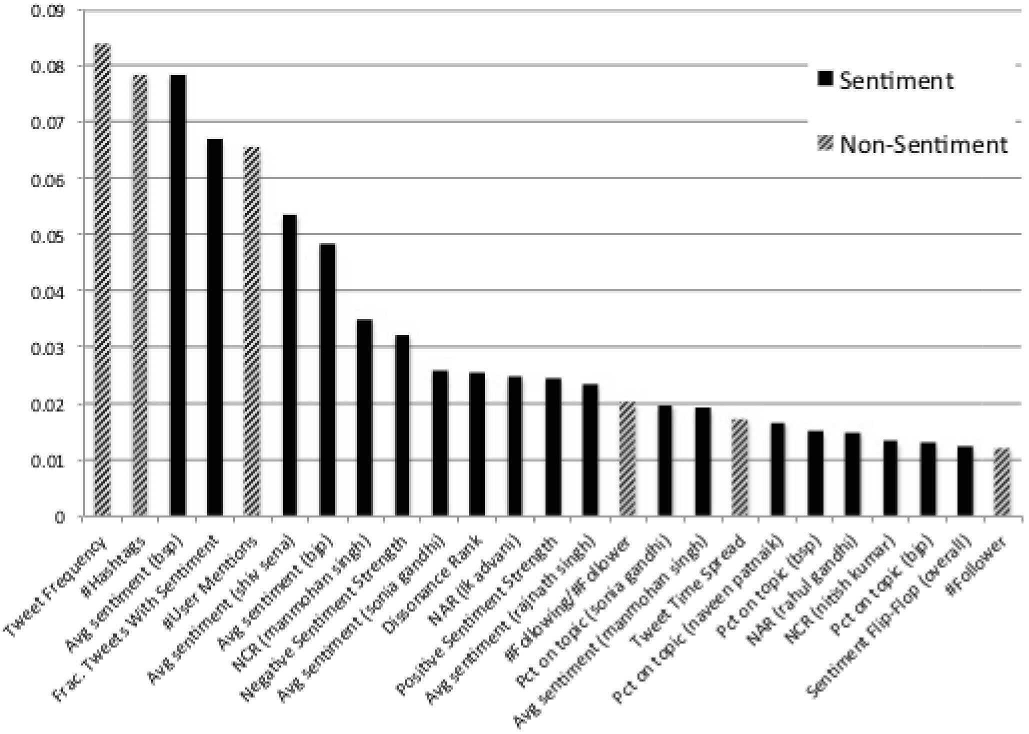


Figure 2.5: Top 25 most important features in the best classifier. Sentiment-based features are shown in black, while standard features are striped [DKS14].

2.2.2 Conclusion on the Effectiveness of Sentiment Analysis for Fake News Detection and Approaches

As highlighted in the preceding Subsection, sentiment analysis can be leveraged as a supplementary feature in the fake news detection process. This involves performing feature engineering and integrating the sentiment of the text as an additional column in the dataset.

Sentiment analysis can be performed using machine learning methods, which typically involve several general steps such as data collection, data preprocessing, feature extraction, model training, model evaluation, and prediction. However, in this thesis, the goal is not limited to applying a single fixed classification algorithm for model training. Instead, the aim is to employ an ensemble technique for sentiment classification. The ensemble technique involves training multiple machine learning models using different methods. These models are trained on the preprocessed data and learn to classify the sentiment of the text. Rather than relying on a single model, an ensemble approach combines the predictions of these models (base learners or base classifiers) to make a final decision. To achieve this, a voting classifier is employed, which allows each model in the ensemble to vote for the sentiment prediction. The voting classifier takes into account the predictions from all the models and selects the sentiment label that receives the most votes as the final result.

Kazmaier et al. [KVV22] provide valuable insights into the primary reasons why ensembles can outperform single classifiers in the context of sentiment analysis. They explain that ensembles benefit from the concept of diversity, which refers to the use of multiple models that are distinct from each other in terms of their learning algorithms, feature representations, or training data.

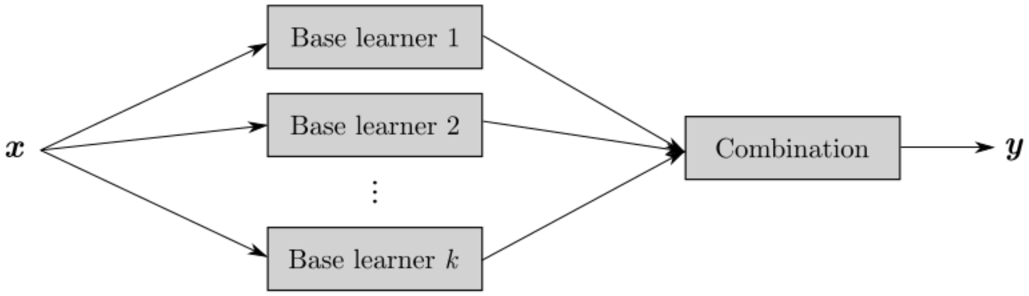


Figure 2.6: A typical ensemble architecture [KVV22].

2.3 Exploring Potential Domains for Test Data Selection in Fake News Detection

Based on the majority of relevant publications, the potential domains for fake news detection have been categorized into four main categories. The "Politics" category includes applications that focus on analyzing fake news in relation to political topics, such as the presidential election or political campaigns. The "Health" category encompasses the detection of fake news in the domain of diseases, conditions, and overall health, including areas like COVID-19 misinformation. The "Climate Change" category involves fake news detection research specifically related to climate change and its various aspects. Lastly, the "Social Media" category comprises publications primarily focused on analyzing fake news in relation to world news topics discussed on social media annotated either manually or crowd-sourced. Listed below are some of the notable publications and datasets that have emerged in these respective categories.

2.3.1 Politics

Poddar et al. [PU+19] conducted a comprehensive comparative analysis of various machine learning models for fake news detection. Their analysis involved utilizing a dataset comprised of articles that shared political affiliations. This enabled them to evaluate the effectiveness of different models in discerning the veracity of news content within a political context.

In a similar vein, Wang et al. [Wan17] curated the LIAR dataset by collecting political statements from the fact-checking website PolitiFact[3]. This dataset has been widely utilized as a valuable resource for evaluating the performance of fake news detection models specifically in the domain of political discourse.

Additionally, the ISOT[4] fake news dataset was curated by the University of Victoria ISOT Research Lab. It consists of both truthful news articles sourced from Reuters and fake news articles obtained from unreliable websites that have been flagged by Politifact. This dataset provides researchers with a valuable resource to investigate and develop effective fake news detection models.

2.3.2 Health

Cui et al. [CL20] introduced the COAID dataset, which was specifically developed for detecting healthcare misinformation related to COVID-19. The dataset encompasses fake news gathered from websites and social platforms, alongside users' social engagement with such news. The researchers employed various fake news detection methods utilizing different algorithms and conducted an evaluation of the dataset. Notably, the convolutional neural network (CNN) achieved a precision of 96.53% in their evaluation.

In a similar vein, Cheng et al. [CWY+21] compiled both news articles and Twitter data to construct a COVID-19 rumor dataset. The dataset incorporates a comprehensive set of features, including stance, sentiment, and veracity.

[3] https://www.politifact.com/ (*Last access as of August 3, 2023*)

[4] https://onlineacademiccommunity.uvic.ca/isot/2022/11/27/fake-news-detection-datasets/ (*Last access as of August 3, 2023*)

2.3.3 Climate Change

Diggelmann et al. [DBGB+20] contributed to the field by introducing the CLIMATE-FEVER dataset, which serves as a valuable resource for verifying climate change-related claims. This dataset enables researchers to explore and develop models that can effectively analyze and evaluate the veracity of information pertaining to climate change.

In a similar context, Meddeb et al. [MRD+22] presented a novel dataset comprising over 2300 articles written in French, collected through web scraping from various media sources focusing on climate change. They employed a BERT-based model and achieved a notable F1-score of 84.75% in evaluating the dataset's performance. This dataset provides researchers with an opportunity to investigate and address the challenges associated with climate change misinformation in the French language domain.

2.3.4 Social Media

Shu et al. [SMW+20] made a significant contribution to the field by creating a comprehensive data repository called FakeNewsNet. This repository, available on GitHub, offers a wide range of features that facilitate the study of fake news on social media platforms. Researchers can utilize this valuable resource to investigate various aspects of fake news, including its spread, impact, and detection methods.

Additionally, several datasets have been curated to address the specific challenges associated with fake news and rumor detection on social media platforms. The BuzzFace[5] and FacebookHoax[6] datasets were collected from Facebook comments related to news articles and labeled based on the veracity of the news. These datasets enable researchers to explore the effectiveness of different techniques in identifying fake news within the context of user discussions on social media.

The CREDBANK[7] dataset, on the other hand, was specifically gathered to evaluate the credibility of tweets on Twitter. By analyzing this dataset, researchers can develop and test models to assess the credibility of information shared on the popular microblogging platform.

Qazvinian et al. [QRRM11] conducted a study focused on rumors and collected a large dataset consisting of tweets about specific rumors within a certain time period on Twitter. They explored the effectiveness of three categories of features: content-based, network-based, and microblog-specific memes, in accurately identifying rumors. This dataset provides valuable insights into the characteristics and dynamics of rumor propagation on social media platforms.

Furthermore, Starbird [Sta17] collected Twitter data to analyze and describe the emerging alternative media ecosystem. The study aimed to provide insights into how websites promoting conspiracy theories and pseudo-science may function to serve underlying

[5] https://github.com/gsantia/BuzzFace (*Last access as of August 3, 2023*)

[6] https://github.com/gabll/some-like-it-hoax (*Last access as of August 3, 2023*)

[7] http://compsocial.github.io/CREDBANK-data/ (*Last access as of August 3, 2023*)

political agendas. The Twitter data collected in this study offers valuable information for understanding the role of social media in the dissemination of misinformation and its potential impact on public opinion.

2.4 Performance Evaluation of Fake News Detection Solutions

The evaluation of a classification algorithm's performance necessitates the utilization of meaningful metrics that facilitate robust generalization [BOSB10]. As emphasized in Table 1.2 of Subsection 1.4.2, particularly within the context of a multiple case study, it is imperative to employ suitable metrics for assessing the cases [VST+09]. To ensure the validity of these metrics, they must adhere to the requirements outlined in Table 1.3 as outlined by Verner et al. [VST+09].

Upon analyzing the test data, it was observed that imbalances exist in the distribution of predictive characteristics within the dataset, which is subsequently employed for evaluating the proposed solutions. The term "imbalanced" denotes an uneven distribution of classes to be predicted within the dataset. This scenario is a common occurrence in real-world scenarios [TAS+20] and often leads to a bias favoring the more prevalent classes during the training of classification models [BOSB10]. Additionally, even when evaluating pre-trained models, imbalances within the dataset can yield dissimilar valid outcomes. Consequently, it is imperative to meticulously consider these imbalances when selecting appropriate performance metrics for evaluation purposes.

During the literature review conducted on performance metrics for multiclass classification, accuracy, precision, recall, and F1 score were identified as performance indicators for evaluating the outcomes. The reasons behind the selection of each metric will be described in subsequent Subsections.

Accuracy

Together with the error rate, accuracy is the most widely used metric to evaluate the performance of classification solutions [TAS+20]. The calculation is carried out according to formula 2.1 based on the ratio of the number of correct predictions to the number of all predictions.

$$Accuracy = \frac{TruePositive + TrueNegative}{TruePositive + TrueNegative + FalsePositive + FalseNegative} \tag{2.1}$$

This metric has a notable limitation in that it does not consider the distribution of classes within the dataset [KEP17]. Consequently, the accuracy value can be heavily influenced by the model's ability to predict the most prevalent class [GBV20]. To obtain a more comprehensive and nuanced assessment of performance, additional metrics are gathered throughout the evaluation process. These metrics aim to provide more understanding of the model's performance across various aspects.

Precision

Precision is a metric that evaluates the accuracy of positive predictions made by a model. It addresses the question of what proportion of positive identifications was actually correct. In other words, precision measures the percentage of correctly predicted positive instances out of all the instances that were predicted as positive [GBV20]. The calculation of precision is performed using the following formula:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{2.2}$$

Precision provides an insight into the model's ability to avoid false positive errors. High precision in fake news detection indicates that the model has a low rate of false positives, meaning it accurately identifies genuine instances of fake news. It is important to minimize false positives in this domain to prevent the incorrect labeling of legitimate news articles as fake. A high precision value indicates that when the model flags an article as fake, it is highly likely to be accurate [KKK20]. This metric has some limitations that should be considered. It does not consider the false negatives. Moreover, it can be affected by class imbalance, where one class significantly outweighs the other in terms of the number of instances. In such cases, the model may achieve high precision by predicting the majority class accurately, while ignoring or misclassifying instances from the minority class [TAS+20]. To overcome these limitations, additional evaluation metrics such as recall, F1 score, accuracy have to be considered.

Recall

Recall is a metric used in classification to assess a model's ability to identify all relevant instances of a particular class. It answers the question: What proportion of actual positive instances was correctly identified? It is also known as sensitivity or true positive rate, measures the percentage of correctly predicted positive instances out of all the actual positive instances [GBV20]. The calculation of recall is performed using the following formula:

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{2.3}$$

In the context of fake news detection, recall measures the model's ability to correctly identify as many instances of fake news as possible out of all the actual fake news instances present in the dataset. A high recall value indicates that the model has a low rate of false negatives, meaning it is effective at capturing a large portion of the fake news articles [KKK20]. However, similar to precision, recall should not be considered in isolation and it is essential to consider the trade-offs between recall and other metrics, such as precision, accuracy, and F1 score for a better understanding of model's performance.

F1-Score

The F1-score is a commonly used metric in classification tasks, including fake news detection, that combines the precision and recall metrics into a single value. It provides a balanced measure of a model's performance by taking into account both the ability to correctly identify positive instances (precision) and the ability to capture all positive instances (recall) [GBV20]. The calculation is carried out according to formula 2.4.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.4}$$

The F1-score is calculated as the harmonic mean of precision and recall, and it ranges from 0 to 1. A high F1-score indicates a good balance between precision and recall, meaning the model has both a low rate of false positives and a low rate of false negatives.

The F1 score is particularly useful when the dataset is imbalanced, meaning one class (e.g., fake news) is much more prevalent than the other (e.g., genuine news). In such cases, optimizing for accuracy alone may not be sufficient, as a model that always predicts the majority class would achieve a high accuracy but fail to capture the minority class effectively. The F1-score takes into account both the false positives and false negatives, providing a more comprehensive evaluation of the model's performance [TAS+20].

CHAPTER 3

Implementation of Fake News Detection Solutions

All implementations in this study were carried out using the Python programming language.

3.1 Selection of Source Test Data

During the literature research several domains, namely „Politics“, „Health“, „Climate Change“, and „Social Media“ were identified as highly relevant areas for fake news detection. Subsequently, publicly available datasets pertaining to each domain were collected and thoroughly analyzed. The Table in Appendix 1 provides the links to these datasets for reference. In cases where multiple datasets met the criteria, the dataset with the largest volume of data was selected to ensure a broader and more representative set for fine-tuning and evaluation purposes. Figure 3.1 presents an overview of the selected datasets. The subsequent Sections will provide brief descriptions of these datasets, including examples of the texts contained within them and their corresponding classes. Furthermore, a quantitative analysis of each dataset will be presented, providing an understanding of their characteristics.

Dataset	Domain	Source	Number of Records
ISOT Fake News	Politics	Truthful Articles From Reuters.com Fake Articles From Unreliable Websites Flagged by Politifact	44898
CoAID	Health (COVID-19)	Social Media Health Organizations Websites Such as WHO, CDC and NIH	127538
CLIMATE-FEVER	Climate Change	Web Scraping Annotation Based on the Majority Vote of Evidences Retrieved From Wikipedia	1535
Community Notes	Social Media (World News)	Twitter Posts Classified Based on The Majority Vote of Notes Done by Community Contributors.	45036

Table 3.1: Datasets overview

3.1.1 Dataset Characteristic – „Politics“

The selected test dataset for the "Politics" category consists of texts written in English, with a specific emphasis on political subjects. This dataset is labeled as either real or fake, and it was initially curated by the University of Victoria ISOT Research Lab. The lab collected the truthful articles by crawling content from Reuters.com, a reputable news website. On the other hand, the fake news articles were gathered from unreliable sources that were flagged by Politifact fact-checking organization and Wikipedia. The dataset encompasses various types of articles covering a range of topics, but the majority of them revolve around political news.

The dataset contains 44898 records and no null values. Table 3.2 contains an example record per label. However, due to the nature of the dataset containing lengthy news articles, only the first two or three sentences of each record have been displayed.

Text	Label
„MOSCOW (Reuters) - President Vladimir Putin said on Wednesday that ties with U.S. President Donald Trump s administration were not without problems, but he hoped that mutual interests of Russia and the United States in fighting terrorism would help improve Moscow s relations with Washington.“	Real
„Fox News host Sean Hannity spent a whopping 6 seconds covering Tuesday night s election results in which Democrats gained seats across the country in a referendum to Donald Trump and his policies. Hannity, an ardent defender of Trump s, explained the Democratic electoral victories in three of the states by saying, Those results in Virginia, New Jersey, and New York not states Donald Trump won.“	Fake

Table 3.2: Dataset example – Politics

3.1.2 Dataset Characteristic – „Health“

The selected test dataset for the "Health" category, namely CoAID (Covid-19 heAlthcare mIsinformation Dataset), comprises a comprehensive collection of news articles and claims specifically related to COVID-19, all written in English. The dataset is categorized as either real or fake, with its creation attributed to the work of Cui et al. [CL20] from the Pennsylvania State University. In their research, they emphasize the distinction between news articles and claims, with claims being notably shorter and typically consisting of one or two sentences, such as "Eating garlic prevents infection with the new coronavirus."

To gather reliable news articles, the researchers employed web crawling techniques on nine reputable media websites. Additionally, they collected tweets from the official Twitter accounts of these websites, further ensuring the inclusion of reliable information. The selected media websites include Healthline, ScienceDaily, NIH (National Institutes of Health), MNT13 (Medical News Today), Mayo Clinic, Cleveland Clinic, WebMD, WHO (World Health Organization), and CDC (Centers for Disease Control and Prevention).

In order to compile fake news and claims, the researchers retrieved URLs from several reputable fact-checking websites, such as LeadStories, PolitiFact, FactCheck.org, CheckYourFact, AFP Fact Check, and Health Feedback. Subsequently, they collected news articles and claims from these identified websites, ensuring the inclusion of diverse sources. Furthermore, they collected fake tweets related to the gathered news articles and claims, ensuring that these tweets underwent thorough fact-checking by the aforementioned reliable sources.

Additionally, the researchers gathered real and fake news and claims from four prominent social media platforms: Facebook, Instagram, Youtube, and TikTok. Similarly, these social media posts were subjected to fact-checking by the aforementioned reliable sources. Furthermore, the researchers collected user engagement features of Twitter tweets, including users' replies to each tweet, adding an additional dimension to the dataset.

The dataset contains 300943 records and no null values. Tables 3.3 contain an example record per label.

Text	Label
„Experts recommend everyone avoid large gatherings especially those with people outside your social bubble during the july fourth holiday weekend. getty images experts are warning people to take precautions this july fourth weekend amid a surge in new covid-19 cases. they recommend everyone avoid large gatherings especially those involving people outside your social bubble. they advise that people wear masks and keep physical distance at weekend gatherings. they also recommend that tab. ‚experts recommend everyone avoid large gatherings especially those with people outside your social bubble during the july fourth holiday weekend.“	Real
„Antibodies for the common cold produce a positive COVID-19 test. False-positive results from COVID-19 antibody testing are behind the COVID-19 cases reported in the U.S.“	Fake

Table 3.3: Dataset example – Health (COVID-19) News

3.1.3 Dataset Characteristic – „Climate Change“

The chosen test dataset for the "Climate Change" category, namely climate-fever, was curated by Diggelmann et al. [DBGB$^+$20] from the University of Zurich's Center of Competence for Sustainable Finance. This dataset was specifically assembled for the purpose of verifying climate change-related claims written in English. To compile a comprehensive set of candidate climate claims from the Internet, the researchers followed an ad-hoc approach and employed a predefined set of seed keywords for targeted Google searches.

The claims were obtained through a combination of manual retrieval and automated web scraping techniques. Each claim in the dataset is accompanied by five carefully annotated votes, with supporting evidence sentences retrieved from Wikipedia. These evidence sentences serve to either support, refute, or provide insufficient information to validate the claim. In cases where both supporting and refuting evidence were found, the claim-label is assigned as „DISPUTED“. The final verdict for each claim is determined based on a majority vote. By default, the claim-label is set as „NOT_ENOUGH_INFO“ unless there is clear supporting evidence ("SUPPORTS") or refuting evidence („REFUTES“). If there is both supporting and refuting evidence the claim-label is DISPUTED.

The dataset contains 1535 records and no null values. Table 3.4 contains an example record per label.

Text	Label
„One of the main areas of contention is the existence of two strange climate episodes known as The Medieval Warm Period (MWP) and the Little Ice Age.“	SUPPORTS
„The warming is not nearly as great as the climate change computer models have predicted.“	REFUTES
„Although the extent of the summer sea ice after 2006 dropped abruptly to levels not expected until 2050, the predicted 67-per-cent decline in polar bear numbers simply didn't happen.“	DISPUTED
„It's not carbon dioxide, it's not methane... Scientists estimate that somewhere between 75% and 90% of Earth greenhouse effect is caused by water vapor in clouds.“	NOT_ENOUGH_INFO

Table 3.4: Dataset example – Climate Change

3.1.4 Dataset Characteristic – „Social Media“

The selected test dataset for the "Social Media" category is the Community Notes Dataset, which serves as a crowd-sourced world news-related database derived from Twitter tweets. The fundamental principle behind this dataset is that each tweet on Twitter has the potential to receive notes, which are contributions made by individuals on the platform. These notes are displayed alongside the tweets only if they have been deemed helpful by other contributors with diverse perspectives. Furthermore, the dataset incorporates a classification label assigned through a majority vote of the notes, categorized as either NOT_MISLEADING or MISINFORMED_OR_POTENTIALLY_MISLEADING. The decision to select this dataset was driven by its reliance on crowd intelligence, which effectively integrates human intelligence into the realm of text analytics.

The dataset contains 45036 records and no null values. Table 3.5 contains an example record per label.

Text	Label
„Eggs used to be a cheap protein. Now they're practically a delicacy. We've got to pump the breaks on the overspending in Washington in order to get inflation under control.“	NOT_MISLEADING
„Every single Republican in Congress voted against capping out-of-pocket drug costs for seniors at $2,000 a year.“	MISINFORMED_ OR_ POTENTIALLY_ MISLEADING

Table 3.5: Dataset example – Social Media

3.1.5 Next Step: Enrichment of Selected Datasets

Among the selected datasets, there are some that include Twitter tweet data, which currently only consist of tweet IDs. In order to utilize these datasets effectively, it is necessary to extract the corresponding tweet texts from Twitter. Furthermore, as discussed in Subsection 2.2.2 of the literature review, incorporating sentiment analysis as an additional feature has the potential to enhance the effectiveness of fake news detection. To integrate this feature, the final datasets will be constructed by appending a sentiment column to the original selected datasets. Therefore, before delving into the quantitative analysis of the test data, the following Sections will provide a detailed explanation of the processes involved in preparing an intermediary dataset by extracting tweet texts from Twitter (only for the datasets containing Twitter data). Subsequently, the sentiment of the extracted texts will be predicted and added as additional columns to build the final dataset.

The datasets exhibits variations in their structures, requiring the development of separate Python modules for the preparation of each final dataset. These dedicated modules were designed to handle the specific characteristics and requirements of each dataset.

3.2 Building the intermediary Datasets: Extracting Tweet texts from Twitter

The implementation of this step involved utilizing the Tweepy library in Python to access the Twitter API. To access the Twitter API and utilize the Tweepy module in Python, it is necessary to create a Twitter developer account. However, during the testing phase, it became clear that in order to retrieve tweets older than the last seven days, one must not only have „Essential“ access/account but also upgrade to „Academic Research“ or „Elevated“ access levels, which provide the ability to search the full archive of Twitter tweets and offer greater flexibility. The Twitter developer account access provides the necessary credentials and tokens to execute API calls. Moreover, as the original dataset files exclusively consisted of tweet IDs, it was necessary to develop an efficient and time-saving dataset-building module in Python. This involved creating a list of all tweet IDs and passing them as a batch to the Twitter API to retrieve the corresponding tweet texts collectively. The `statuses_lookup()` method available in the Tweepy module's API class proved to be useful in obtaining the contents associated with each tweet ID, commonly referred to as a „status“, which encompasses both the metadata and the text of the tweets. However, it is important to note that this function has a limitation of processing only up to a hundred tweet IDs or status IDs at a time. Consequently, the original dataset, in CSV format, had to be divided into smaller chunks of CSV files, each containing only a hundred rows, to accommodate the limitations of the function. Lastly, to ensure consistency in the feature names for all respective datasets and facilitate their utilization in the subsequent sentiment analysis and fake news detection processes, the tweet texts were incorporated as the column labeled „text“ in each dataset and saved a new CSV file for further use.

3.3 Building the Final Datasets: Incorporating Sentiment Analysis as an Additional Feature

With the availability of the „text“ column in all datasets, the subsequent task involved the addition of a complementary column named "sentiment" to each dataset, which represents the predicted sentiment for each corresponding text entry.

3.3.1 Building the Required Modules for Sentiment Analysis

The implementation of this step drew upon the NLTK tutorials available on PythonProgramming.net[1] which provided valuable guidance and insights into the practical aspects of working with NLTK for sentiment analysis.

As outlined in Subsection 2.2.2 of the literature review, this thesis endeavors to employ ensemble learning techniques for sentiment analysis. To achieve this, a dedicated module was developed, based on the sentiment analysis tutorial[2] and utilizing both „Scikit-Learn“ and „NLTK“ libraries. This module encompasses the training, testing, and serialization of various classifiers, allowing them to be saved using the „Pickle“ library. This ensures their preservation for future utilization in the sentiment analysis and fake news detection processes.

- Naive Bayes
- Bernoulli Naive Bayes
- Multinomial Naive Bayes
- Linear Support Vector Machine
- Logistic Regression
- Stochastic Gradient Descent Classifier
- Maximum Entropy

Furthermore, an additional module was developed to perform the classification of a text’s sentiment. This module is based on the tutorial on „Combining Algorithms with NLTK“ tutorial[3]. The module incorporates a voting system, where each algorithm is allocated one vote, and the classification that receives the highest number of votes is selected as the final prediction. To implement this functionality, the module creates a custom classifier class named VoteClassifier, inheriting from the NLTK classifier class. In the subsequent step, the module iterates through the list of saved classifiers. For each classifier, it classifies the requested text and treats the classification as a vote. After completing all iterations, the module returns the most popular vote, representing the prediction of the best-performing classifier.

[1] `https://pythonprogramming.net/` (*Last access as of August 3, 2023*)

[2] `https://pythonprogramming.net/new-data-set-training-nltk-tutorial/` (*Last access as of August 3, 2023*)

[3] `https://pythonprogramming.net/combine-classifier-algorithms-nltk-tutorial/` (*Last access as of August 3, 2023*)

Both constructed modules encompass text pre-processing steps aimed at optimizing the data for sentiment analysis and enhancing the performance of the classifiers. These pre-processing steps consist of various tasks, including the handling of special characters, text tokenization, removal of stopwords, and lemmatization. In this particular step, lemmatization was preferred over stemming. Lemmatization aims to identify the base form of words, such as transforming 'troubled' to 'trouble', whereas stemming may result in incorrect meanings and spelling errors, such as converting it to 'troubl', leading to linguistically inaccurate results.

As final step, the selection of a reliable labeled dataset was required to train sentiment classification models. The initial approach involved constructing binary sentiment analysis classifiers using the Large Movie Review Dataset[4], curated by Maas et al. [MDP+11] from Stanford University. Movie reviews are one of the most prevalent and commonly used domain for sentiment analysis, as they provide a rich source of text data with clear sentiment expressions. The dataset consists of 50,000 reviews sourced from IMDB, ensuring an equal number of positive and negative reviews. The aforementioned classifiers were trained on this meticulously labeled dataset and subsequently evaluated for their performance.

Table 3.6 presents two exemplary texts extracted from the test set, offering insights into their actual sentiments, predicted sentiments, and the corresponding accuracies. Notably, the second review in the table exhibits an instance where the sentiment prediction was inaccurate. This review encompasses both negative and positive sentences, highlighting a challenge encountered by binary sentiment classifiers when confronted with dual-polarity sentences, as discussed in the Sentiment Analysis Subsection 1.1.3. Such sentences can pose difficulties for binary classifiers, leading to erroneous predictions. Consequently, it is crucial to acknowledge the limitations of binary classification and its potential failure in certain scenarios.

Considering these challenges, the sentiment classification approach was extended to encompass three sentiment classes: Positive, Neutral, and Negative. To facilitate this expansion, the financial phrase bank dataset was selected and downloaded from the „Huggingface.com“ platform. This dataset comprises nearly 5000 English sentences extracted from financial news, with each sentence meticulously annotated by domain experts. The new classifiers were constructed employing the same classification algorithms as previously mentioned, and they replaced the previous classifiers.

[4] https://ai.stanford.edu/~amaas/data/sentiment/ (*Last access as of August 3, 2023*)

Text	**Actual Sentiment**	**Predicted Sentiment**	**Accuracy**
„The cast played Shakespeare. Shakespeare lost. I appreciate that this is trying to bring Shakespeare to the masses, but why ruin something so good. Is it because 'The Scottish Play' is my favorite Shakespeare? I do not know. What I do know is that a certain Rev Bowdler (hence bowdlerization) tried to do something similar in the Victorian era. In other words, you cannot improve perfection. I have no more to write but as I have to write at least ten lines of text (and English composition was never my forte I will just have to keep going and say that this movie, as the saying goes, just does not cut it.“	Negative	Negative	85.71%
„Encouraged by the positive comments about this film on here I was looking forward to watching this film. Bad mistake. I've seen 950+ films and this is truly one of the worst of them - it's awful in almost every way: editing, pacing, storyline, 'acting,' soundtrack (the film's only song - a lame country tune - is played no less than four times). The film looks cheap and nasty and is boring in the extreme. Rarely have I been so happy to see the end credits of a film. The only thing that prevents me giving this a 1-score is Harvey Keitel - while this is far from his best performance he at least seems to be making a bit of an effort. One for Keitel obsessives only.“	Negative	Positive	53.82%

Table 3.6: Sentiment of movie reviews predicted as the result of the best performing binary classifier.

3.3.2 Construction of the final datasets

The construction of the final datasets involved the utilization of a sentiment classifier module. This module facilitated the integration of sentiment analysis as an additional feature in the datasets. To achieve this, the text of each row in the datasets was passed through the sentiment classifier module, and the predicted sentiment was subsequently added as a new column to each dataset.

3.4 Quantitative Analysis of Test Data

The datasets have a consistent structure. The first column of the dataset contains the text associated with topics within the field, written in English. The second column is dedicated to assigning a sentiment label to each text, indicating the sentiment expressed in the text. Additionally, depending on the domain, a third column is included to represent the specific class associated with each text. The tables presented in the first Section of this chapter provide an overview of the assigned classes for each domain.

In this Section of the study, a quantitative analysis of the test data is conducted. The purpose of this analysis is to gain deeper insights into the data and facilitate subsequent pre-processing steps. Visual representations are utilized to illustrate the frequencies of individual words, the distribution of truthfulness labels across the entire datasets, as well as the distribution of truthfulness labels within each sentiment category. These graphical representations provide a clear and concise overview of the distribution patterns.

3.4.1 Quantitative Analysis of Data from the Field „Politics“

To gain insights into the textual content being analyzed, Figure 3.1 demonstrates the word frequency distribution within the dataset. Stop words have been excluded from the analysis, as they are typically eliminated during the pre-processing phase. The findings reveal a clear connection to the domain of politics, emphasizing the presence of relevant terms and highlighting the contextual relevance of the dataset.

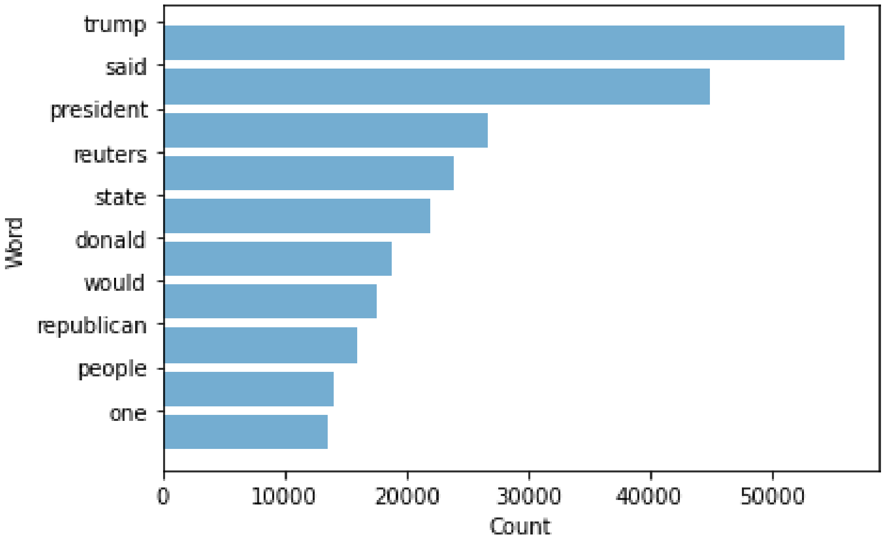


Figure 3.1: Most common words – Politics

The distribution of the labels is visualized in Figure 3.2. The figure illustrates the frequency of each label within the dataset. It can be observed that the dataset exhibits a nearly balanced distribution across the different labels.

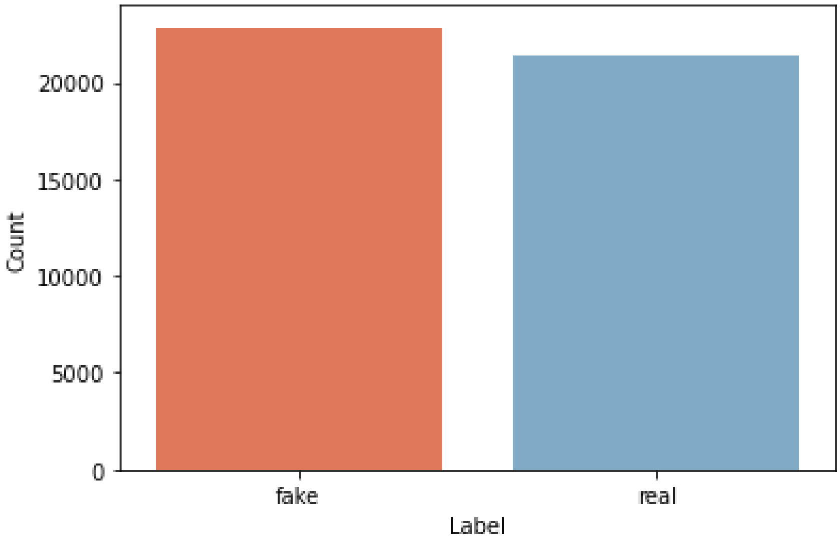


Figure 3.2: Distribution of dataset labels – Politics

Furthermore, Figure 3.3 provides insights into the distribution of each label per sentiment category. It can be observed that the distribution of sentiment categories for each label is nearly similar.

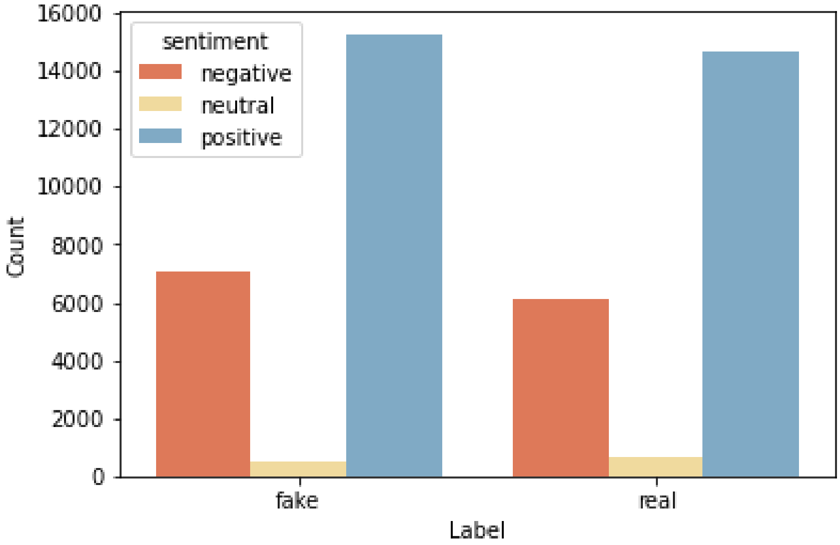


Figure 3.3: Distribution of dataset labels per sentiment category – Politics

3.4.2 Quantitative Analysis of Data from the Field „Health (COVID-19)“

Figure 3.4 presents an overview of the most frequent words in the Health (COVID-19) dataset. The analysis reveals that a considerable portion of the text content is focused on the topic of coronavirus, whereas discussions related to general health are represented approximately half as frequently.

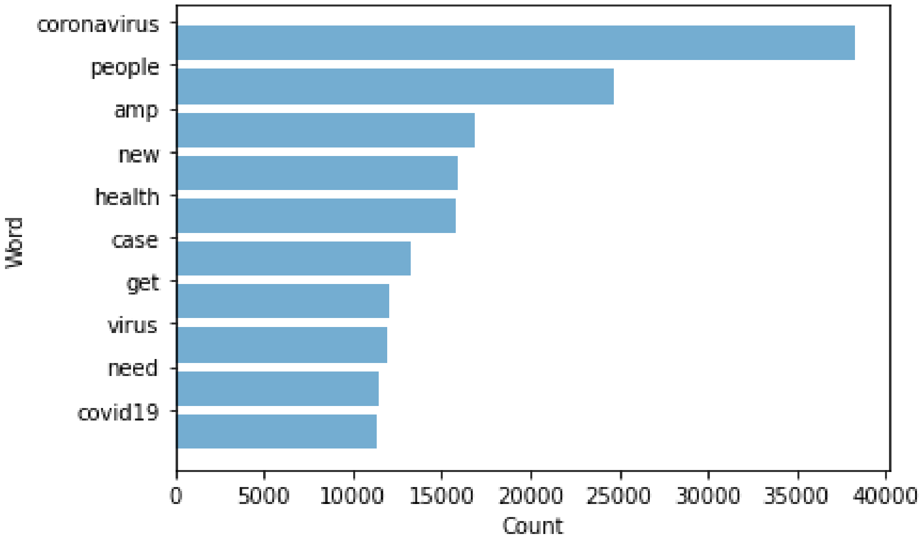


Figure 3.4: Most common words – Health

The distribution of each label within the dataset is visualized in Figure 3.5. Despite the significant class imbalance in the dataset, it was still chosen due to its extensive usage in several publications.

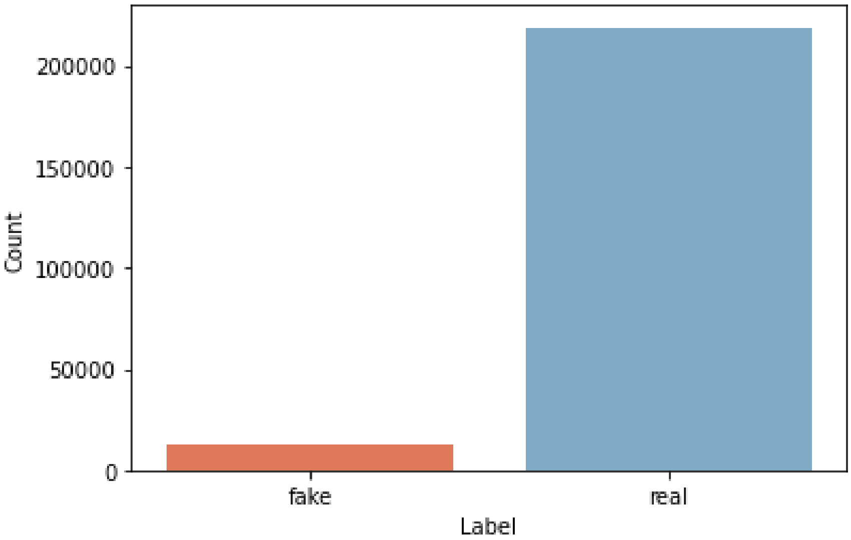


Figure 3.5: Distribution of dataset labels – Health

Figure 3.6 displays the distribution of sentiment categories within the dataset.

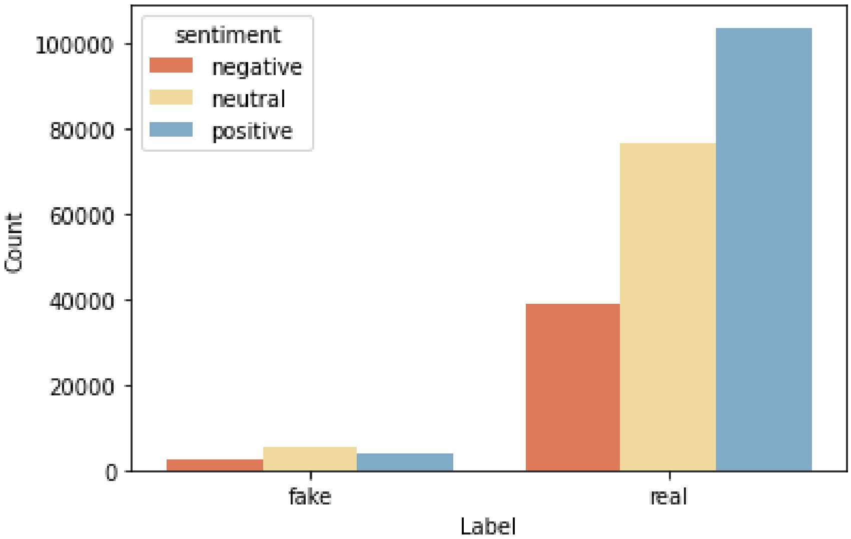


Figure 3.6: Distribution of dataset labels per sentiment category – Health

It can be observed that real news/tweets have a tendency to be more positive than negative. On the other hand, the distribution of sentiment categories in fake news/tweets is relatively balanced, with a slight inclination towards neutrality.

3.4.3 Quantitative Analysis of Data from the Field „Climate Change“

Figure 3.7 demonstrates the word frequency distribution within the dataset. It is observable that the majority of the conversations in the dataset revolve around the subject of global warming.

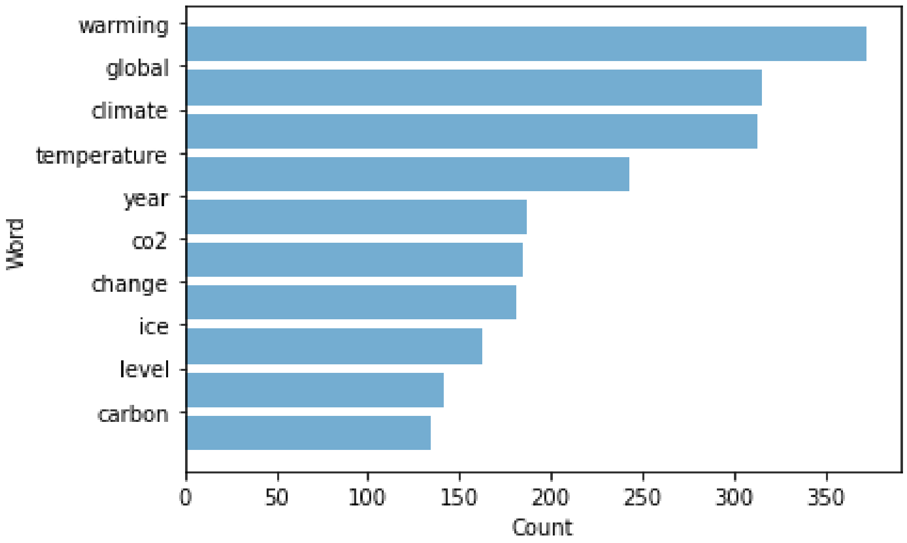


Figure 3.7: Most common words – Climate Change

The distribution of labels is illustrated in Figure 3.8. Despite the small size of the dataset and the significant imbalance in label distribution, it was selected for analysis due to its utilization in various previous publications. Moreover, this dataset contains the available collection of short texts focusing on climate change supports and denials.

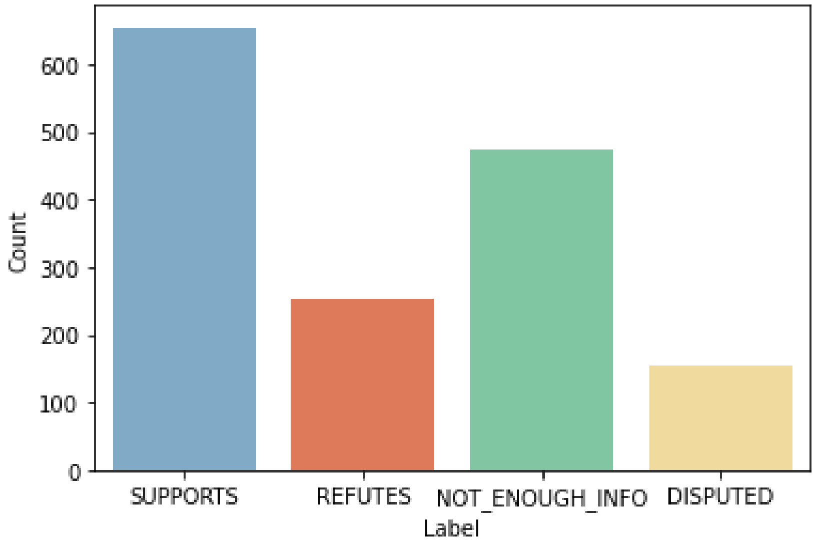


Figure 3.8: Distribution of dataset labels – Climate Change

Furthermore, Figure 3.9 provides insights into the distribution of each label per sentiment category. Similar to the health (COVID-19) dataset´, it is evident that discussions supporting climate change tend to exhibit a more positive sentiment than negative. On the other hand, the distribution of sentiment categories in rejection claims is not predominantly negative but rather leans towards neutrality, with a slight inclination towards positivity.

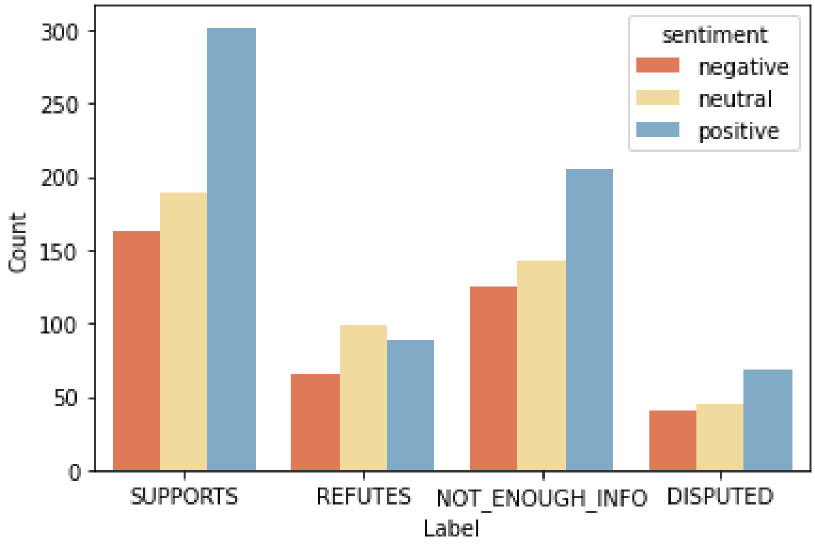


Figure 3.9: Distribution of dataset labels per sentiment category – Climate Change

3.4.4 Quantitative Analysis of Data from the Field „Social Media“

Figure 3.10 provides an overview of the most frequently occurring words in the Twitter's Community Notes dataset. The analysis indicates that the tweets cover a wide range of general world news topics.

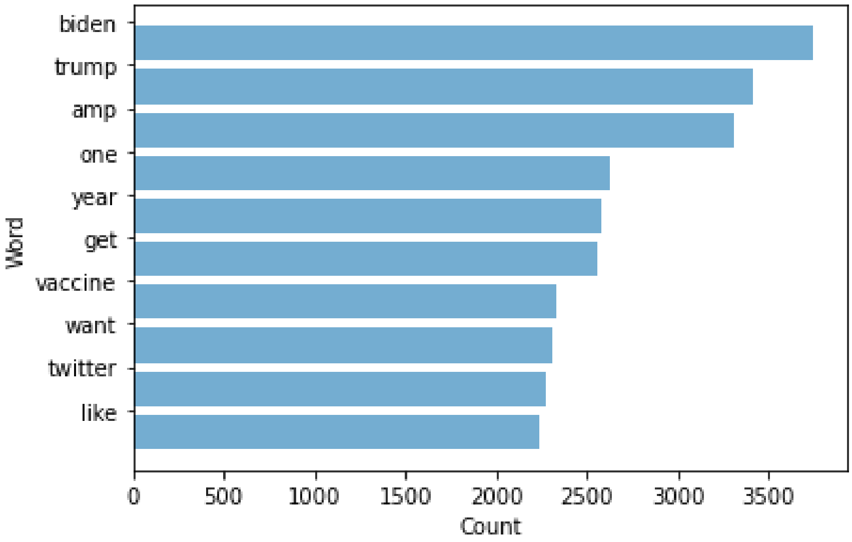


Figure 3.10: Most common words – Social Media

The distribution of each label within the dataset is visualized in Figure 3.11.

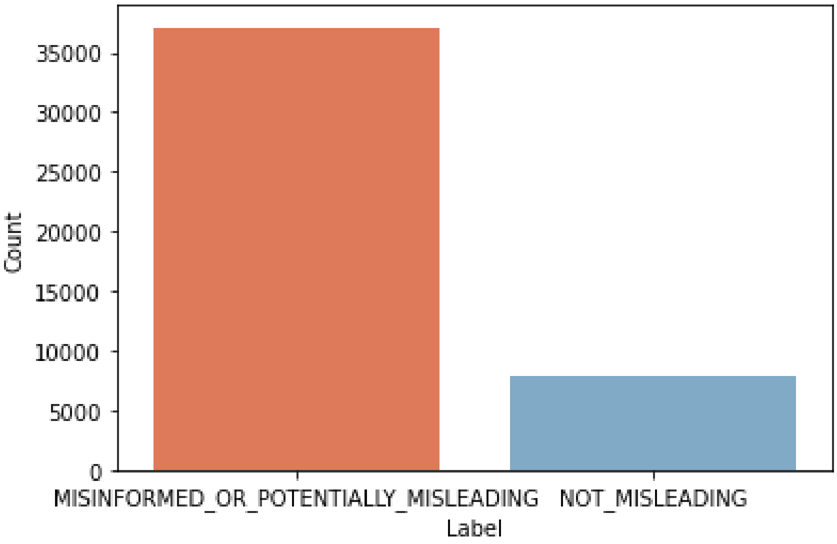


Figure 3.11: Distribution of dataset labels – Social Media

While misinformation tweets predominate, it is notable that only about a fourth of the tweets in the dataset are not misleading. This significant imbalance in the distribution of labels should be considered during the performance evaluation and interpretation of the specified metrics.

Figure 3.6 displays the distribution of sentiment categories within the dataset.

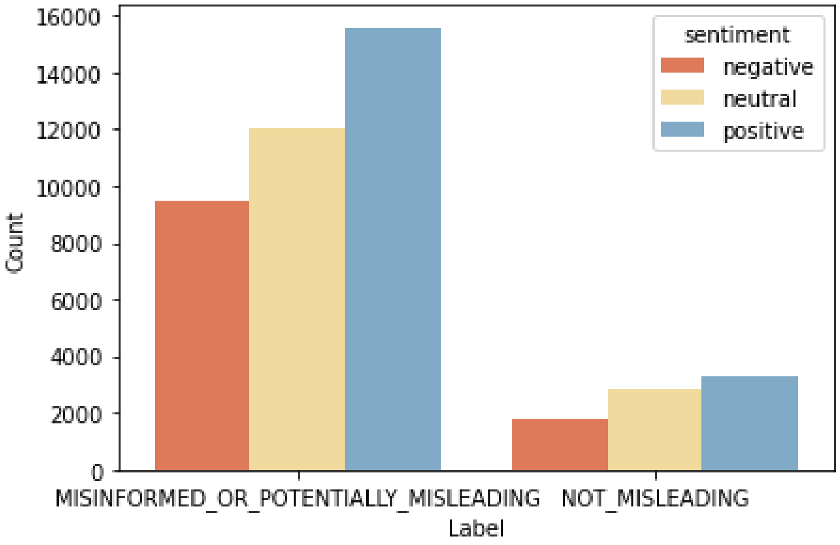


Figure 3.12: Distribution of dataset labels per sentiment category – Social Media

It can be observed that similar to the politics dataset the distribution of sentiment categories for each label is nearly similar.

3.5 Pre-Processing of Datasets

The process of data pre-processing is depicted in Figure 3.13, illustrating the sequence of steps involved in transforming the raw text data into a refined and standardized format. An individual module was developed to encompass all the steps discussed in this Section.

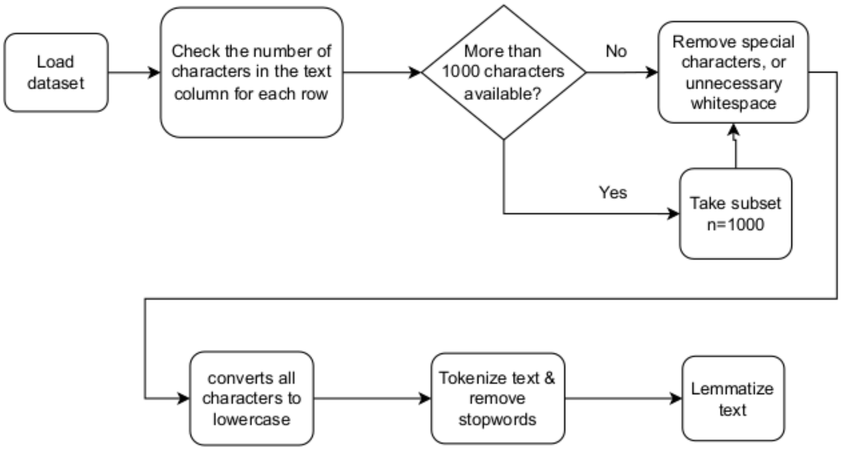

Load dataset
Check the number of characters in the text column for each row
More than 1000 characters available?
No
Remove special characters, or unnecessary whitespace
Yes
Take subset n=1000
converts all characters to lowercase
Tokenize text & remove stopwords
Lemmatize text

Figure 3.13: Pre-processing procedure

To mitigate computational costs and training time, a preliminary decision was made to limit the analysis to the first 1000 characters of the full article text or social media post. Subsequently, a data cleaning process was applied, which involved removing special characters and unnecessary whitespace. To standardize the text, all characters were converted to lowercase. Tokenization was then employed to segment the texts into individual units, and NLTK library-provided stopwords were removed to exclude less informative words, such as articles, pronouns, and prepositions. Finally, a lemmatization technique was employed to reduce words to their base form.

3.5.1 Preparing Dataset Classes and Labels

Once the preprocessing of the datasets is complete, and they have been standardized to a consistent format, the next step is to prepare the dataset classes and labels to meet the requirements for training machine learning and neural network models. This involves the subsequent steps.

Numerical Encoding of Labels

In this preprocessing step, the categorical labels present in each dataset, such as „fake/real“ or „MISINFORMATION/NOT-MISLEADING“, were converted to numerical format. This conversion was necessary due to the requirement of numerical inputs by machine learning algorithms during the training and prediction phases.

Vectorization of Text Features

The subsequent step involved the transformation of textual data into numerical representations that are comprehensible and processable by machine learning algorithms. There are several common approaches for text vectorization such as TF-IDF Vectorizer, CountVectorizer, and Word Embeddings such as Word2Vec, GloVe, and FastText. The choice of technique depends on the specific use-case, and there is no definitive answer as to which one is superior. The key distinction lies in the fact that TF-IDF Vectorizer and CountVectorizer convert sentences into vectors, while Word2Vec converts individual words into vectors [TAR16].

For this study, the chosen method of text vectorization was CountVectorizer utilizing only bigrams. This technique tallies the occurrences of words in articles or posts and subsequently transforms the text into a matrix of tokens. As a result, it not only captures word frequencies but also incorporates tokenization. The output of this process yields a sparse matrix representation.

Concatenation of Multiple Features

In order to make the classification model take into account the two columns text and sentiment, there were two options to consider. The first option involved utilizing Scikit-Learn library's Pipeline and FeatureUnion to select and handle multiple columns, including text, numerical, or binary features. Alternatively, the second option was to generate two sets of features independently and subsequently concatenate the two feature vectors. The pre-processing module implemented the second approach. This way the learning algorithm can assign different weights to each independent feature.

Since the text column had already been transformed into a sparse matrix in the preceding step, enabling concatenation required converting the values in the sentiment column to numerical values for each row, Figure 3.14.

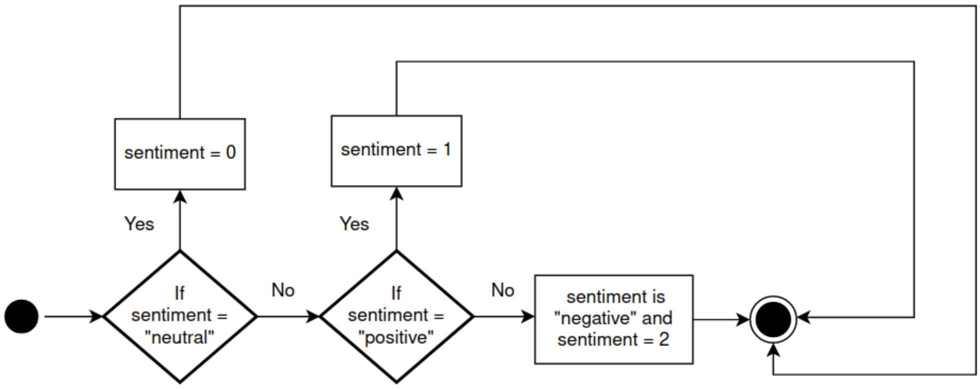

sentiment = 0
sentiment = 1
Yes
Yes
If sentiment = "neutral"
No
If sentiment = "positive"
No
sentiment is "negative" and sentiment = 2

Figure 3.14: Converting sentiments to numerical values.

Afterwards, the sentiment values were converted into a NumPy array. The module then utilized the Sparse hstack function from the SciPy library to concatenate the two vectors/matrices of features.

3.6 Fake News Detection Using Supervised Machine Learning and Deep Learning Algorithms

The selection of models employed in this study was based on the literature review, as detailed in Section 2.1. However, for the sake of completeness, the following Subsections present a brief overview of each algorithm along with their key characteristics and features. Subsequently, the experimental setup will be explained, encompassing the methodology employed to conduct the experiments. Subsequently, the implementation Section will be presented, providing an explanation of the implementation process.

3.6.1 Models and Characteristics

In this Section, the chosen models for the task will be discussed along with their respective properties.

Support Vector Machine (SVM)

Support Vector Machine (SVM), introduced by Vapnik in 1982, has emerged as a powerful and promising supervised algorithm for data classification. Numerous studies have demonstrated the effectiveness of SVM, particularly in the domain of text classification. According to Wei et al. [WWW12], SVM has shown superior performance compared to traditional methods. However, it is important to note that SVM does have certain limitations. One drawback is its sensitivity to noise during the training phase. Additionally, SVM lacks built-in support for feature selection [WWW12].

The main idea behind the algorithm is to find an optimal hyperplane that best separates data points belonging to different classes in a high-dimensional vector space. The goal is to maximize the margin between the hyperplane and the nearest data points from each class, known as support vectors.

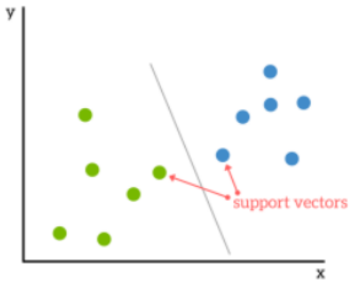


Figure 3.15: A Simple Linear Support Vector Machine [Bam18].

By maximizing this margin, SVM aims to achieve better generalization and improve its ability to classify new, unseen data. In its simplest form, the SVM classifier can be represented by the following formula:

$$f(x) = w^T x + b \tag{3.1}$$

Here, $f(x)$ represents the predicted class label for a given input x. The sign function returns +1 if the expression $w^T x + b$ is positive, indicating one class, and -1 if it is negative, indicating the other class.

The key components of the SVM classifier are:

- w: The weight vector that determines the orientation of the hyperplane.
- x: The input feature vector. In the case of this study x is the concatenation of the text and sentiment features.
- b: The bias term or intercept.

When using a SVM model, there are hyperparameters that require tuning based on the classification problem. One important hyperparameter is the kernel type, which includes options such as Linear, Radial Basis Function, Polynomial, and Sigmoid. Kernels facilitate the transformation of input data to enable the model to more accurately and efficiently determine the decision boundary and classify data points. Another significant hyperparameter is the C parameter, which influences the SVM optimization's emphasis on avoiding misclassification of training examples. A larger value of C leads to the choice of a smaller-margin hyperplane, prioritizing accurate classification of all training points. Conversely, a very small C value prompts the optimizer to seek a larger-margin separating hyperplane, even if it results in the misclassification of some points.

In a study by Altman et al. on fake news detection[5], they experimented with tuning these parameters. They explored different kernel types, and also the various C values, including 0.001, 0.01, 0.1, 1, and 10. The best performing configuration was observed with a linear kernel and a C value of 0.1, resulting in a validation accuracy of 98.07%. Based on these findings, this thesis aims to adopt the same hyperparameter values to achieve comparable outcomes.

[5] https://mehtaplustutoring-mlbootcamp20.github.io/Real_vs_Fake_News/ (*Last access as of August 3, 2023*)

Naive Bayes

Naive Bayes is a commonly employed algorithm in text classification tasks due to its speed, effectiveness, and ease of implementation. It is a basic probabilistic classifier that is based on Bayes' theorem. This classifier has been successfully applied in various domains such as opinion mining, text classification, and spam filtering. The term „naive“ refers to the assumption of feature independence, where each feature is considered independently when classifying the text. The main idea behind Naive Bayes is to calculate the probability of a document belonging to different classes based on the presence of certain words in the document [SZ17].

Naive Bayes classifiers are widely used in various classification tasks, and they come in different variants such as Gaussian Naive Bayes, Multinomial Naive Bayes, and Bernoulli Naive Bayes [SZ17]. The choice of a specific Naive Bayes classifier depends on the characteristics of the features being used. For continuous or binary features, Gaussian or Bernoulli Naive Bayes may be more appropriate, respectively. However, in the context of text classification, Multinomial Naive Bayes is often preferred due to the discrete nature of text data, which can be represented using word frequencies or term frequencies [KHRM06]. The Multinomial Naive Bayes classifier effectively handles this type of data by taking into account the frequency of words or terms in the documents, making it well-suited for tasks such as spam filtering and text categorization [SZ17]. Thus, this thesis employs the Multinomial Naive Bayes classifier for the task of fake news detection.

Given a document represented by a feature vector $\mathbf{x} = (x_1, x_2, ..., x_r)$, where x_i represents the frequency or occurrence of a term or feature i in the document, and a set of classes $C = c_1, c_2, ..., c_k$, the Multinomial Naive Bayes classifier calculates the conditional probability of a document belonging to a particular class.

The formula for calculating the probability of a document $\mathbf{x}$ belonging to a class c is:

$$P(c|\mathbf{x}) = \frac{P(c)\prod P(x_i|c)}{P(\mathbf{x})} \tag{3.2}$$

Where:

- $P(c|\mathbf{x})$is the posterior probability of class c the feature vector $\mathbf{x}$,
- $P(c)$ is the prior probability of class c,
- $P(x_i|c)$ is the likelihood probability of observing the feature x_i given the class c,
- $\prod$ denotes the product operator over all features x_i in the feature vector $\mathbf{x}$, and
- $P(\mathbf{x})$ is the marginal probability of the feature vector $\mathbf{x}$.

To classify a new document, the classifier calculates the probability for each class and assigns the document to the class with the highest probability.

Deep Neural Network

Three prominent models commonly used in deep learning are Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory Neural Networks (LSTMs) [HKS12, LBH15, MGM15, KRSST18]. The choice of the deep architecture depends on the type of data sources. For image sources, CNNs have proven to be a typical and effective architecture [HKS12, LBH15], while RNNs are often utilized for audio sources [MGM15]. On the other hand, LSTM models offer promising performance in capturing dependencies in sequences of words [KRSST18].

CNNs (Convolutional Neural Networks) have gained significant popularity in computer vision tasks, but they can also be effectively used for fake news detection, offering advantages over RNNs (Recurrent Neural Networks) and LSTMs (Long Short-Term Memory). One key advantage of CNNs is their ability to capture local patterns and dependencies in the input data. In the context of text analysis, fake news articles often contain specific word combinations, phrases, or textual patterns that can provide clues about their authenticity. CNNs are suitable at capturing such local features through their convolutional operations, which enable them to identify important linguistic patterns related to fake news. Furthermore, unlike RNNs and LSTMs, which rely on sequential processing and capture long-term dependencies, CNNs are particularly suitable for scenarios where the order of words may not be crucial for determining the authenticity of news articles [KRSST18]. Fake news detection often involves analyzing individual words, phrases, or short textual segments rather than relying heavily on the sequential nature of the text.

A neural network architecture comprises interconnected layers of artificial neurons, also known as nodes or units. Each neuron receives input signals, performs a mathematical operation on them, and generates an output signal that is propagated to the next layer. The connections between neurons possess weighted values, enabling the network to learn and adjust the importance of various features. Considering the aforementioned points, this thesis employs a CNN deep learning model consisting of the following layers:

- Embedding Layer
  This layer associates each word index with an embedding vector. During the training process, the model learns these vectors, optimizing their values based on the given data [Seb02]. In this specific implementation, the embedding layer is configured with a vocabulary size. For this the respective Python module built a dedicated vocabulary for each selected dataset individually. This step was necessary for achieving high accuracy in the model's predictions and managing the computational expenses associated with training the network.

- Dropout layer
  The Dropout Layer was incorporated into the neural network architecture to mitigate the issue of overfitting.

- GlobalAveragePooling1D Layer
  This layer was added to simplify the input data representation by generating a fixed-length output vector. This layer computes the average value across each feature dimension, reducing the dimensionality of the input. By aggregating the information in this way, the model can effectively handle variable-length input data.

- Dense Layer
  Following the GlobalAveragePooling1D layer, the output vector was passed through the Dense layer. The Dense layer is connected to a single output node and applies an activation function. The activation function introduces non-linearity to the model, allowing it to capture complex patterns and make accurate predictions based on the learned features. However, the choice of the appropriate activation function in the neural network model was made based on the characteristics of the dataset. In the Python module, the following activation functions were selected:

  - For binary classification tasks, where only a single output neuron is required, the Rectified Linear Unit (ReLU) activation function was chosen. This activation function is well-suited for handling binary classification problems and was employed for the politics, health, and social media datasets.
  - On the other hand, for multi-class classification tasks, where one output neuron per class is necessary, the Softmax activation function was utilized for the entire output layer. The Softmax activation function is specifically designed to handle multi-class problems and was applied in the case of the climate change dataset.

3.6.2 Experimental Setup

To conduct a comparative case study, a Full-Factorial approach was devised. The Full-Factorial design, illustrated in Figure 3.16, encompasses all possible combinations of 4 datasets, 3 models, and the inclusion or exclusion of sentiment analysis, resulting in a total of 24 distinct combinations. This approach ensures that all factors and their interactions are explored and evaluated. By considering a wide range of combinations, the study aims to provide an understanding of the performance and effectiveness of the selected datasets, models, and the inclusion of sentiment analysis.

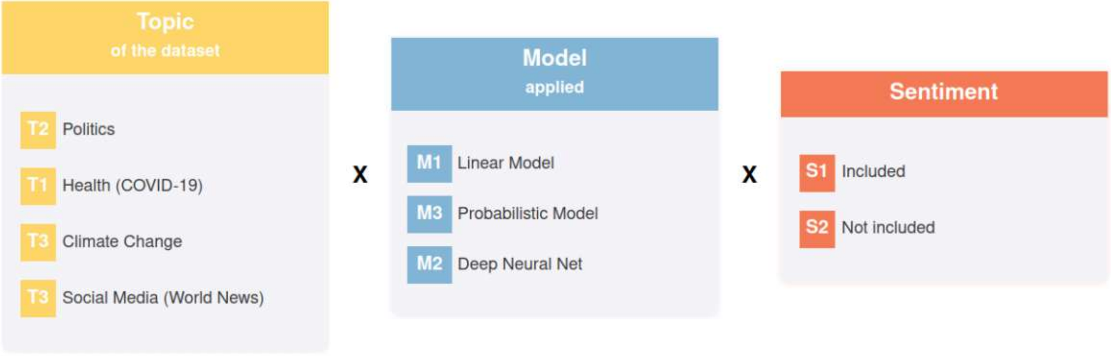


Figure 3.16: Full-Factorial Approach

3.6.3 Implementations

As previously mentioned, the project was implemented in the Python programming language, utilizing supervised learning techniques such as Support Vector Machines (SVM), Multinomial Naive Bayes, and a deep neural network. To facilitate the experimentation process, a dedicated module was developed, encompassing two main methods/functions: „train_model_with_sentiment“ and „train_model_without_sentiment“. These methods accept three arguments, namely path_to_dataset, topic, and model, enabling the exploration of 24 different approaches. The outcomes of each experiment were meticulously recorded and will be presented in the subsequent Section. Considering the arguments of both methods:

- The „path_to_dataset“ argument denotes the path to the dataset CSV file. This CSV file is then read into a dataframe using the Pandas library, facilitating further data processing and analysis.

- The „topic“ argument signifies the topic of the respective dataset, with possible values including "Politics," "Health," "Climate-Change," and "Social-Media." This argument serves multiple purposes, including:

  - Determining the appropriate activation function for the addition of the dense layer in the neural network, as elucidated in the preceding Sections.

  - Guiding the selection of the loss function for compiling the neural network model. For datasets with binary classes, the binary_crossentropy loss function is chosen, whereas for multi-class datasets, the categorical_crossentropy loss function is employed.

- The „model“ argument represents the name of the classifier and can take on values such as „SVM“ for the Support Vector Machine classifier, „NB“ for the Naive Bayes classifier, and „CNN“ for the Convolutional Neural Network.

After passing the required arguments to the methods and calling it the methods will first split the data into training and testing sets with the Scikit-Learn train_test_split function, having a test size of 0.2, indicating that 20% of the data should be held over for testing and 80% for training. Further, it trains the respective selected model with the selected dataset, then it shows the evaluation results. For the machine learning models, SVM and MultinomialNB, the confusion matrix of Scikit-Learn Metrics is applied to show the performance. For the neural network the evaluate() function from Keras which returns two values, namely loss, representing the error and the accuracy. Additionally all performance metrics discussed in Section 2.4 of literature review will be calculated and shown as well.

Upon passing the requisite arguments and invoking the methods, they initiate a series of essential steps. Firstly, the methods segment the data into training and testing sets using the train_test_split function from Scikit-Learn. The test size is set to 0.2, indicating a split where 20% of the data is reserved for testing purposes, while the remaining 80% is allocated for training. Subsequently, the selected model is trained using the chosen dataset, followed by the display of evaluation results.

For the machine learning models, SVM and MultinomialNB, the performance assessment employs the confusion matrix from Scikit-Learn's Metrics module. This matrix provides insights into the model's classification accuracy. In the case of the neural network, the evaluate() function from the Keras library is utilized, returning two values: the loss, representing the error, and the accuracy. Additionally, all relevant performance metrics discussed in Section 2.4 of the literature review will be calculated and presented.

CHAPTER 4

Result Mapping and Performance Evaluation

The implementation steps described in the preceding Chapter were carried out on a per-dataset basis. Considering the politics dataset as an example, initially, all three models were trained, incorporating the sentiment feature of the dataset, utilizing the „train_model_with_sentiment“ method. Subsequently, the same three models were trained once more, this time excluding the sentiment feature, employing the „train_model_without_sentiment“ method. Therefore, the subsequent Sections go through each dataset and report the results obtained for each classifier, accompanied by a thorough performance evaluation. This analysis will be conducted for both cases, with and without the incorporation of sentiment analysis. This provides insights into the effectiveness of the classifiers across different data contexts.

4.1 Assessing Performance in the „Politics“ Category

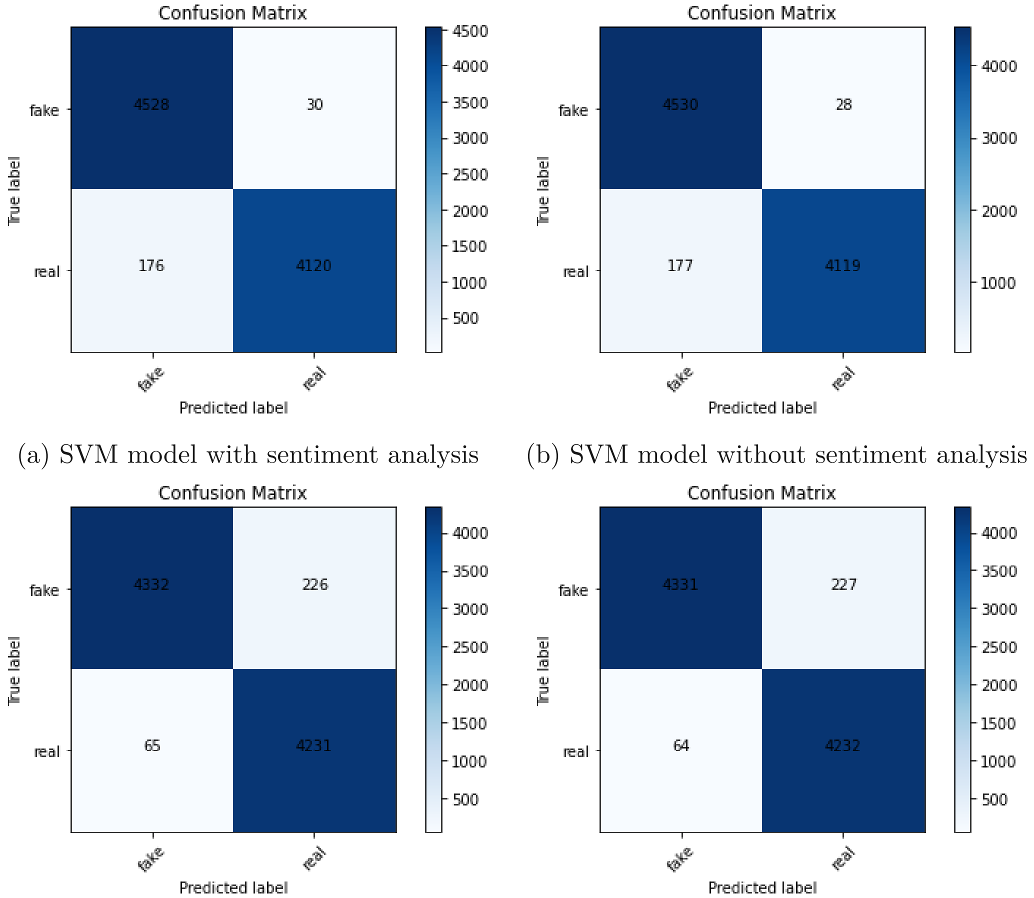
Table 4.1 presents the performance results for the politics dataset.

Model	Sentiment Included				Sentiment Excluded			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
SVM	97.67%	99.28%	95.90%	97.56%	97.68%	99.32%	95.88%	97.57%
NB	96.71%	94.93%	98.49%	96.68%	96.71%	94.91%	98.51%	96.68%
DNN	96.45%	98.33%	96.94%	97.63%	96.93%	98.11%	97.19%	97.65%

Table 4.1: Dataset Performance – Politics

The results demonstrate that the politics dataset achieved the highest performance with the SVM classifier, achieving an accuracy of 97.68%. The deep neural network and naive

Bayes classifiers follow, in that order. Interestingly, incorporating the sentiment feature did not significantly affect the performance of any of the classifiers.



(a) SVM model with sentiment analysis

(b) SVM model without sentiment analysis

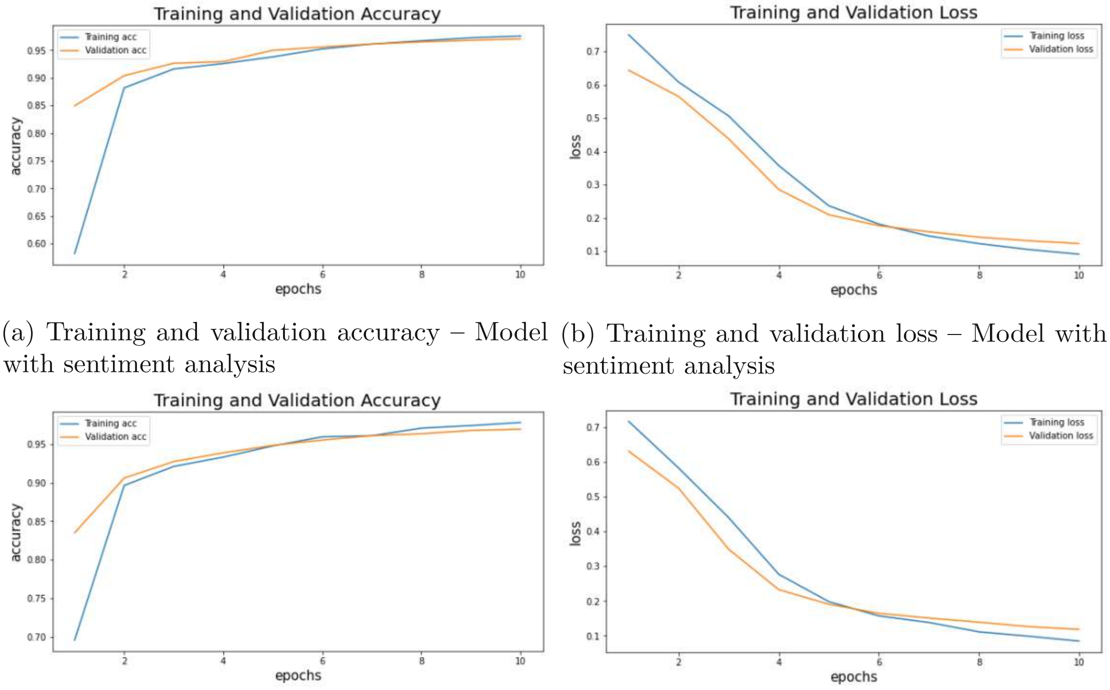
(c) Naive Bayes model with sentiment analysis

(d) Naive Bayes model without sentiment analysis

Figure 4.1: Confusion matrices of trained machine learning models – Politics

Following the examination of the confusion matrices, it is evident that the SVM model accurately classified 4528 instances of fake news as fake and 4120 instances of real news as real when incorporating sentiment. Whereas, when sentiment was excluded, the SVM model correctly classified 4530 instances of fake news as fake and 4119 instances of real news as real. This indicates that the SVM model exhibits better performance in predicting fake news compared to real news and as well slightly better without sentiment analysis.

The performance of the neural network can be observed through the visualization in Figure 4.2. The Figure depicts the progress of accuracy and loss function (Loss) for both models, including and excluding sentiment analysis, utilizing the training and validation data from the politics category.



(a) Training and validation accuracy – Model with sentiment analysis

(b) Training and validation loss – Model with sentiment analysis

(c) Training and validation accuracy – Model without sentiment analysis

(d) Training and validation loss – Model without sentiment analysis

Figure 4.2: Learning curves of neural network models – Politics

As expected, the accuracy of both models exhibits a consistent increase on the training dataset starting from the second epoch, indicating a good fit. This observation is further supported by the decreasing trend of the loss curve, which reflects a reduction in classification errors on the validation dataset. Both models demonstrate a gradual stabilization of both accuracy and loss curves after the sixth epoch. Considering this analysis, the model fine-tuned over six epochs was selected for the final evaluation in both cases.

4.2 Assessing Performance in the „Health“ Category

Taking into account the notable imbalance exhibited by the Health (COVID-19) dataset, as depicted in Figure 3.5 of the Quantitative Analysis of Test Data Section, the initial model training yielded a significantly low performance. However, in order to address this issue, undersampling techniques were employed, resulting in a notable improvement in the accuracy of the models. The neural network model exhibited the best performance, with the accuracy escalating from 34% to 86% with sentiment feature, as presented in

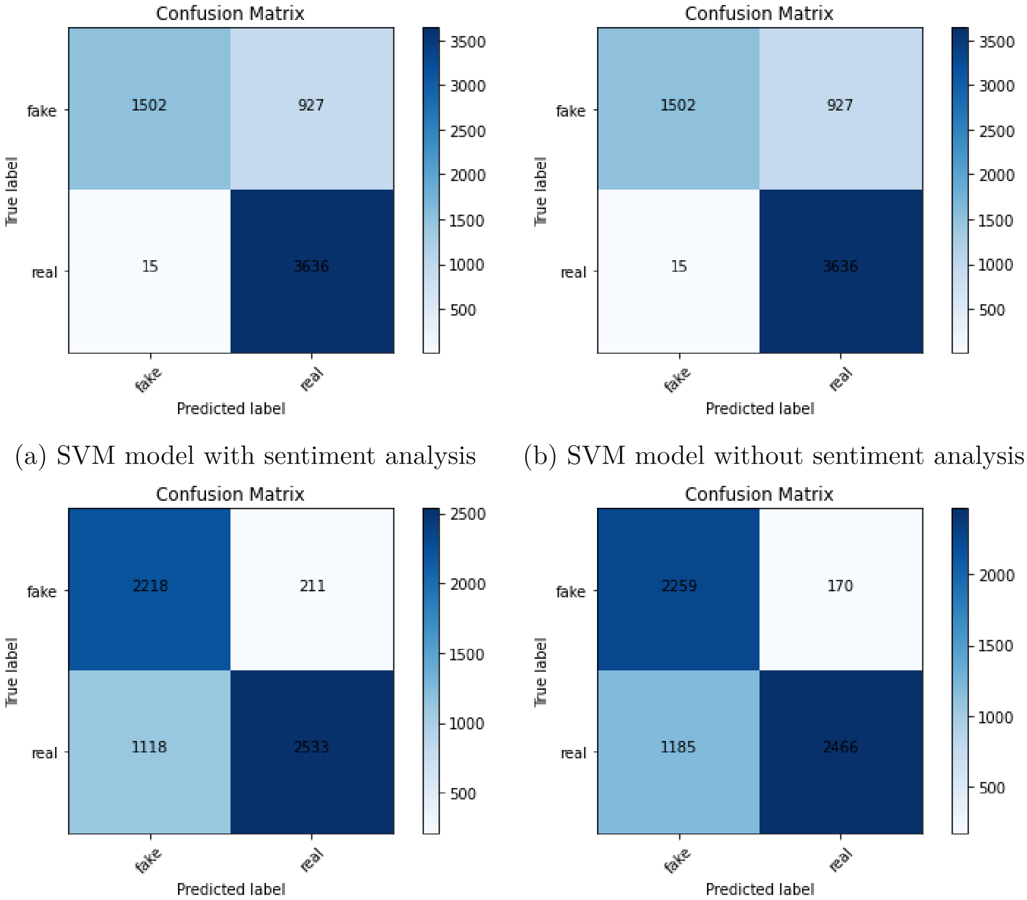
Table 4.2. Undersampling, a technique that involves randomly selecting examples from the majority class, was applied to mitigate the effects of class imbalance.

Model	Sentiment Included				Sentiment Excluded			
	Accuracy	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
SVM	84.51%	79.68%	99.59%	88.53%	84.51%	79.68%	99.59%	88.53%
NB	78.14%	92.31%	69.38%	79.22%	77.71%	93.55%	67.54%	78.45%
DNN	86.23%	84.91%	97.22%	90.65%	80.98%	83.03%	99.46%	90.51%

Table 4.2: Dataset Performance – Health

The findings indicate that the health dataset exhibited the best performance when utilizing the neural network model, particularly when incorporating the sentiment feature. However, upon comparison of all combinations, it is clear that the inclusion of sentiment did not have a significant impact on the classification performance of any of the classifiers.

Furthermore, Table 4.2 indicates that the SVM classifier achieved a higher accuracy of 84.51% compared to 78.14% for Naive Bayes, indicating that it made more correct predictions overall, considering the sentiment analysis included scenario. In terms of precision, which measures the proportion of correctly predicted positive instances among all predicted positive instances, Naive Bayes (92.31%) outperformed SVM (79.68%). This suggests that Naive Bayes had a higher ability to correctly classify true positive instances, this can be further observed by comparing the confusion matrices of the two classifiers as depicted in Figure 4.3. However, when considering recall, which measures the proportion of correctly predicted positive instances among all actual positive instances, SVM (99.59%) performed significantly better than Naive Bayes (69.38%). This means that SVM had a higher capability to identify true positive instances, making it more effective in capturing actual instances of fake news. In summary, in case of sentiment incorporation, the SVM classifier performed better than Naive Bayes, demonstrating higher accuracy, recall, and F1 score, and the neural network performed better than the SVM classifier, demonstrating higher accuracy, precision, and F1 score. In case of sentiment exclusion, the neural network and the SVM classifier behaved almost identical.



(a) SVM model with sentiment analysis

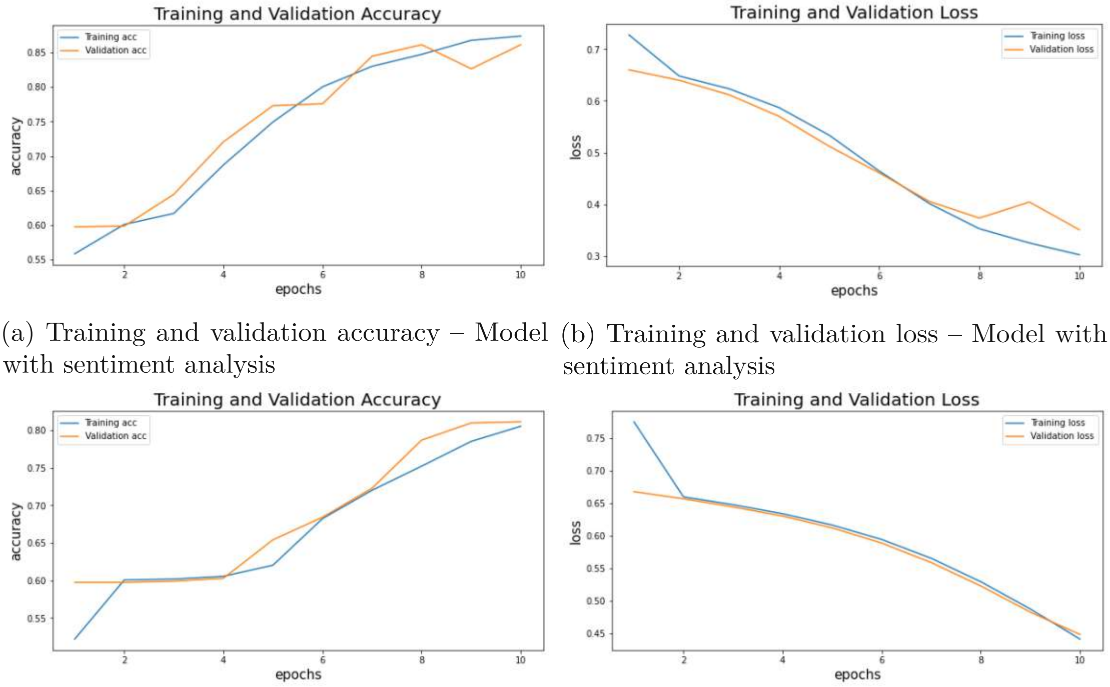
(b) SVM model without sentiment analysis

(c) Naive Bayes model with sentiment analysis

(d) Naive Bayes model without sentiment analysis

Figure 4.3: Confusion matrices of trained machine learning models – Health

The performance of the neural network can be assessed by examining the visualization presented in Figure 4.4. In the case of the model incorporating sentiment analysis, both models demonstrate an initial increase in accuracy on the training dataset. However, the accuracy based on the validation dataset exhibits a noticeable decline starting from the eighth epoch. This decline suggests the occurrence of overfitting, wherein the model becomes excessively tuned to the training data and its ability to generalize to unseen data diminishes. This observation is further supported by the loss history, which reveals an increase in error in the classification of the validation data from the eighth epoch onwards, contrasting with the pattern observed in the training data. On the basis of this analysis, in the case of both scenarios, sentiment included and sentiment excluded, the model that had undergone fine-tuning covering eight epochs was chosen.



(a) Training and validation accuracy – Model with sentiment analysis

(b) Training and validation loss – Model with sentiment analysis

(c) Training and validation accuracy – Model without sentiment analysis

(d) Training and validation loss – Model without sentiment analysis

Figure 4.4: Learning curves of neural network models – Health

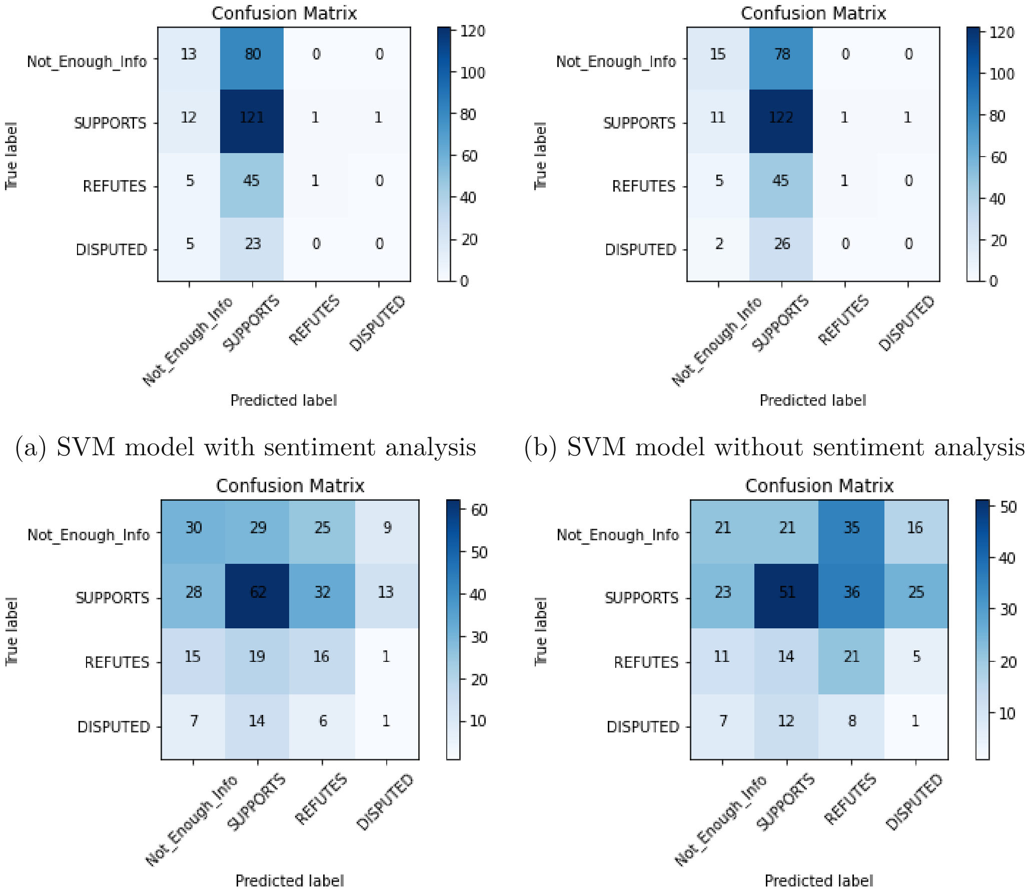
4.3 Assessing Performance in the „Climate Change“ Category

Table 4.3 presents the performance results for the climate change dataset.

Model	Sentiment Included				Sentiment Excluded			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
SVM	43.97%	39.34%	43.97%	33.12%	44.95%	41.87%	44.95%	34.27%
NB	35.50%	37.09%	35.50%	36.00%	30.62%	36.83%	30.62%	32.32%
DNN	44.69%	11.00%	25.00%	15.00%	42.08%	11.00%	25.00%	15.00%

Table 4.3: Dataset Performance – Climate Change

The results indicate that the climate change dataset exhibited a better performance when trained with the SVM classifier without incorporating the sentiment feature, attaining an accuracy of 44.95%. Nevertheless, it is worth noting that overall, the dataset's performance was not ideal across all models assessed.



(a) SVM model with sentiment analysis

(b) SVM model without sentiment analysis

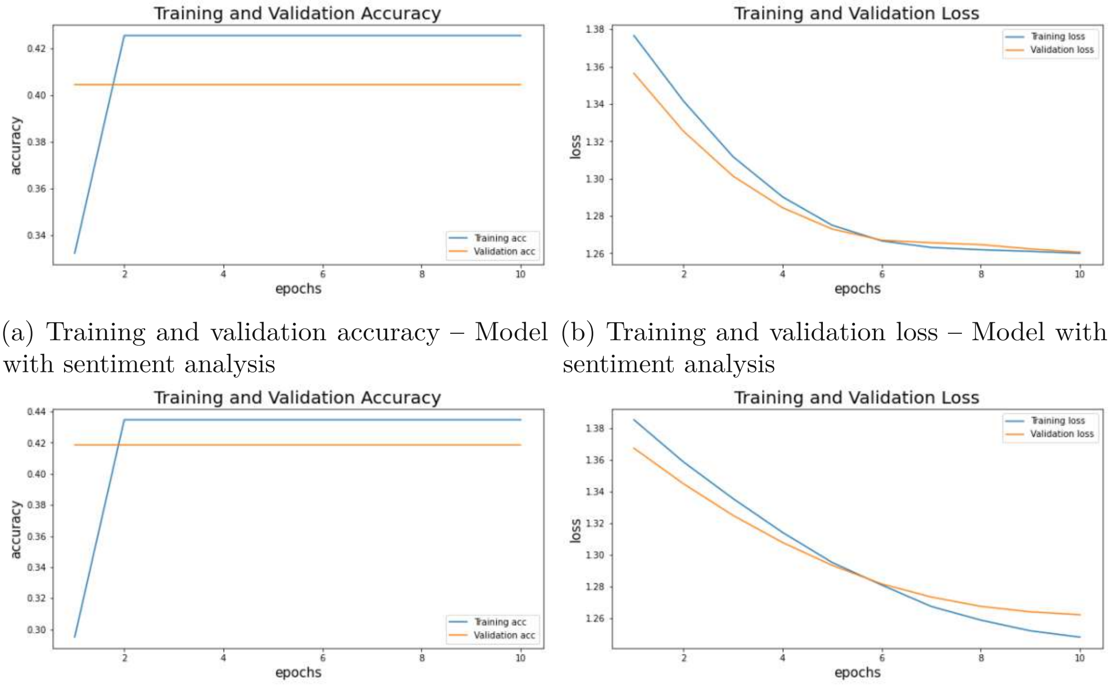
(c) Naive Bayes model with sentiment analysis

(d) Naive Bayes model without sentiment analysis

Figure 4.5: Confusion matrices of trained machine learning models – Climate Change

Following the comparison of the confusion matrices, it is evident that the SVM model accurately classified 121 instances of supporting claims as supporting when incorporating sentiment. Whereas, when sentiment was excluded, the SVM model correctly classified 122 instances of supporting claims as supporting. This indicates that the SVM model exhibits better performance in predicting climate change supporting claims compared to refuting claims and as well slightly better without sentiment analysis.

The performance of the neural network can be observed through the visualization in Figure 4.6. Since the accuracy on both the training and validation datasets remains static, it means that the model is not improving its performance as training progresses. This indicates that the model may have reached a point in its learning process, where further iterations are not leading to significant improvements in accuracy.



(a) Training and validation accuracy – Model with sentiment analysis

(b) Training and validation loss – Model with sentiment analysis

(c) Training and validation accuracy – Model without sentiment analysis

(d) Training and validation loss – Model without sentiment analysis

Figure 4.6: Learning curves of neural network models – Climate Change

In summary, despite achieving the highest accuracy with SVM model among the classifiers, the attained accuracy suggests that further improvements are necessary to enhance the classification performance on the climate change dataset.

4.4 Assessing Performance in the „Social Media“ Category

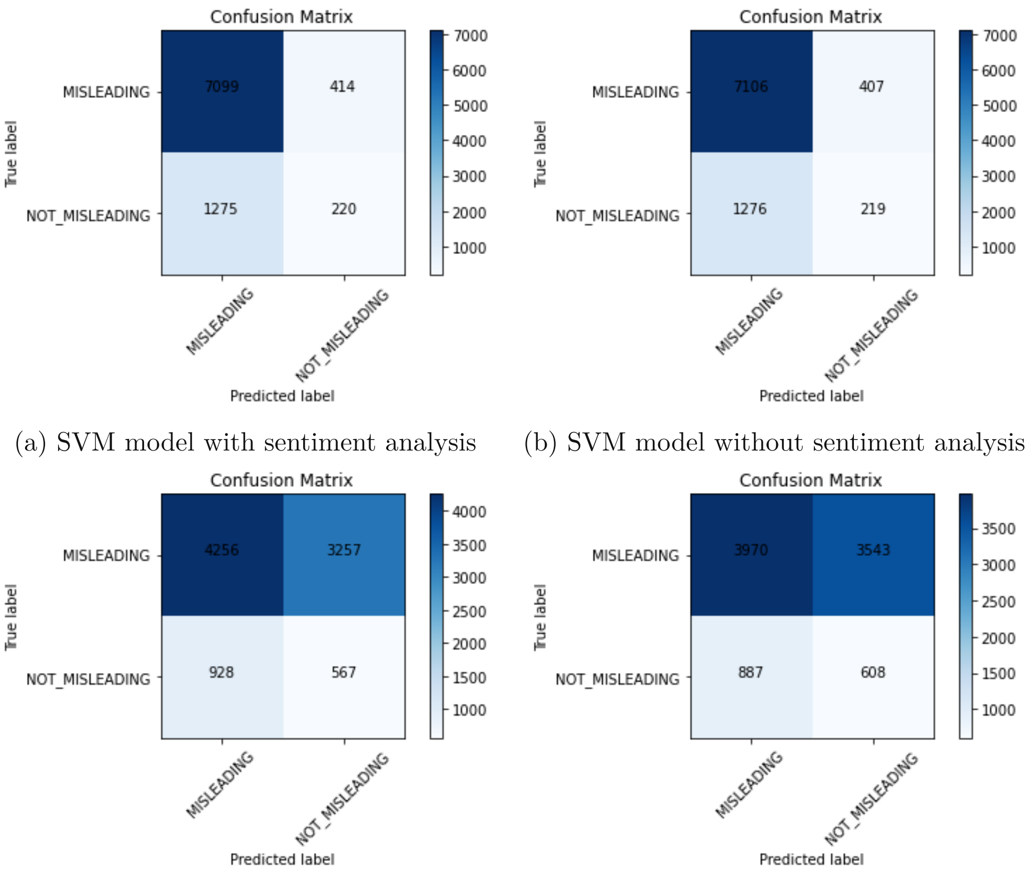
Table 4.4 presents the performance results for the social media dataset.

Model	**Sentiment Included**				**Sentiment Excluded**			
	Accuracy	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
SVM	81.34%	35.24%	14.85%	20.89%	81.32%	34.98%	14.65%	20.65%
NB	53.54%	14.83%	37.93%	21.32%	50.82%	14.65%	40.67%	21.54%
DNN	82.26%	41.00%	50.00%	45.00%	82.18%	41.00%	50.00%	45.00%

Table 4.4: Dataset Performance – Social Media

The findings demonstrate that the neural network exhibited superior performance, attaining an accuracy of 82.26%, when the sentiment feature was included. Nonetheless, the not ideal outcomes observed for other evaluation metrics, including precision, recall, and

F1 score, requires further investigation. It is important to examine the confusion matrices and studying the learning curves to gain insights into the less optimal performance observed in other metrics, such as precision, recall, and F1 score.



(a) SVM model with sentiment analysis

(b) SVM model without sentiment analysis

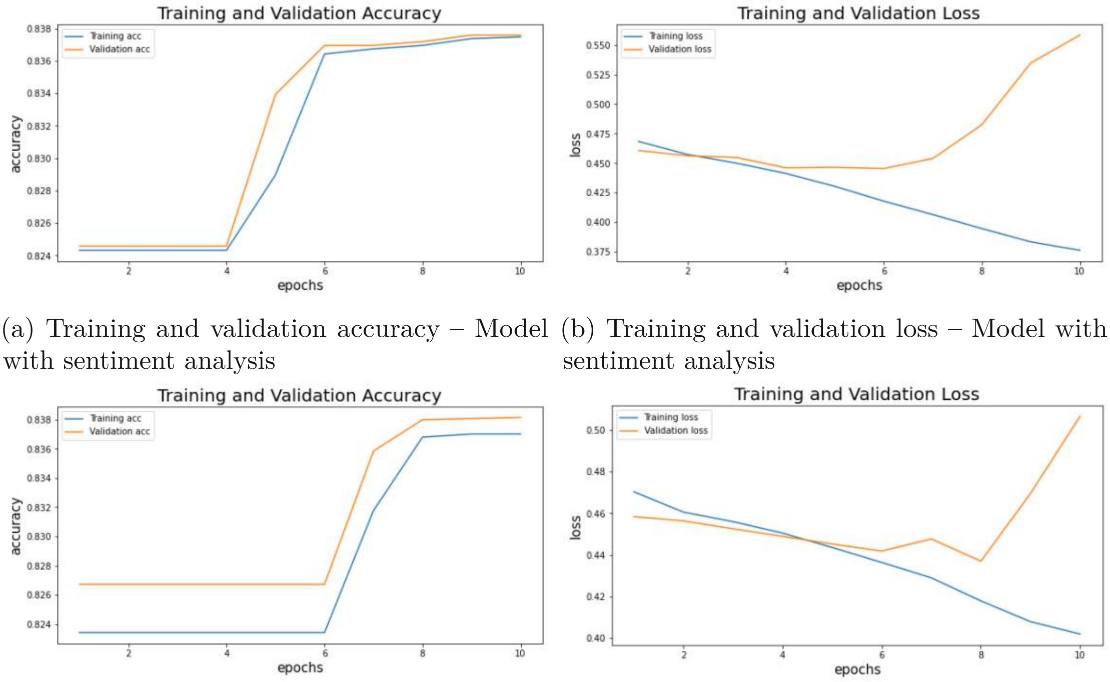
(c) Naive Bayes model with sentiment analysis

(d) Naive Bayes model without sentiment analysis

Figure 4.7: Confusion matrices of trained machine learning models – Social Media

The confusion matrices of SVM models show that the model correctly predicts instances of fake news as fake, but struggles to correctly classify instances of real news as real, it indicates a specific type of error known as a false negative. This means that the model is more prone to incorrectly classifying real news as fake, leading to a higher number of false negatives in the predictions. The same behavior can be seen with the Naive Bayes classifier.

The performance of the neural network can be observed through the visualization in Figure 4.8.



(a) Training and validation accuracy – Model with sentiment analysis

(b) Training and validation loss – Model with sentiment analysis

(c) Training and validation accuracy – Model without sentiment analysis

(d) Training and validation loss – Model without sentiment analysis

Figure 4.8: Learning curves of neural network models – Social Media

The accuracy of the model with sentiment analysis remained static after six epochs, while the loss function was increased continuously. This observation suggests the occurrence of overfitting, indicating that the model became excessively specialized to the training data and encountered challenges in generalizing effectively to unseen data. Consequently, a decline in performance on the validation data was observed. Choosing a six-epoch training for the final model is a way to mitigate overfitting and strike a balance between model performance and generalization. The aforementioned observation holds true for the model when sentiment analysis is not incorporated.

CHAPTER 5

Discussion

5.1 Examining Results in the Context of Previous Literature

Comparing the results with other benchmarking studies enables the identification of certain patterns. Additionally, Table 5.1 is added which compares the training durations of the models across different datasets, in order to assist drawing generalized conclusions. Khan et al. [KKA+21] conducted a similar evaluation focused on different datasets and methods, however, without considering different domains. Interestingly, their findings highlight Naive Bayes as the best-performing traditional machine learning model, particularly suitable when hardware constraints are a consideration. This study aligns with their results, as Table 5.1 shows, where Naive Bayes demonstrates the shortest training time. In another comparison study by Poddar et al. [PU+19], they examined various fake news detection approaches but only on a single dataset, with different sample sizes. Their findings suggest that Support Vector Machine (SVM) performs better with larger datasets, which is consistent with the observations in this study. However, the contextual impact of different datasets from diverse domains was not considered in their analysis. Similarly, Aphiwongsophon et al. [AC18] focused solely on the social media domain, specifically Twitter tweets. While their work provided insights into fake news detection in the context of social media, they did not explore other domains. Interestingly, their findings also indicate that among the selected models, Naive Bayes yielded lower performance measures, while SVM and Neural Network exhibited equivalent results. Gupta et al. [GMBG] exclusively explored traditional machine learning models and did not include any deep learning approaches.

The implementation of fake news detection using diverse approaches and datasets in this study has emphasized the significant impact of a well-established dataset on the overall performance. Specifically, the analysis of the politics dataset has revealed notable improvements. Beyond classification performance, the findings shed light on an additional aspect that is unrelated to performance itself. In an era where computing capacity is billed based on usage costs, the results unveil an important consideration. It is evident that the training time required for the models is not necessarily proportional to the achieved performance. Notably, the Naive Bayes model demands significantly less computational effort compared to SVM and Neural Network models, making it an efficient choice in terms of resource utilization.

Dataset	Model	Total Duration with SA [S]	Total Duration without SA [S]
Politics	SVM	00:31:32	00:30:05
	NB	00:03:11	00:02:53
	DNN	00:07:09	00:06:34
Health	SVM	00:04:55	00:04:45
	NB	00:01:16	00:01:10
	DNN	00:05:52	00:04:26
Climate Change	SVM	00:00:04	00:00:02
	NB	00:00:03	00:00:02
	DNN	00:00:13	00:11:10
Social Media	SVM	00:11:10	00:10:03
	NB	00:02:18	00:02:02
	DNN	00:10:10	00:06:25

Table 5.1: Comparison of the model training durations.

5.2 Limitations

Several limitations exist within this study that necessitate consideration in the overall assessment. Firstly, it is important to note that the selected textual datasets exhibit variations in terms of text length and quality [Kie19]. This inherent heterogeneity poses challenges in directly comparing prevalent fake news domains, as the differing characteristics of the datasets may introduce biases and impact the performance evaluation and generalizability of the results. However, it is important to note that finding labeled datasets covering all domains from a single source, such as exclusively from a social network or a news website, proved to be challenging.

Secondly, the incorporation of sentiment analysis in the fake news detection process primarily considered the overall sentiment type of the texts, such as positive, negative, or neutral. However, a more comprehensive feature engineering approach could be explored, as discussed in the work of Dickerson et al. [DKS14], to incorporate additional sentiment-based features. These features could include average topic sentiment, polarity fractions, positive sentiment strength, and negative sentiment strength, which may further enhance the accuracy and robustness of the detection model.

Thirdly, it is important to acknowledge that this study focuses solely on content-based features for fake news detection and does not consider context-based features such as user networks and user behavior. One major obstacle in this regard was the lack of diversity in the existing datasets. For example, user-related features were not obtainable from the dataset of news articles or website claims. Exploring and incorporating context-based features could provide valuable insights into the social dynamics and user interactions surrounding the spreading of fake news.

By acknowledging these limitations, this study strives to provide a well-informed analysis of fake news detection, while also highlighting areas for further research and improvement.

CHAPTER 6

Conclusion

6.1 Overall Comparison

Table 6.1 provides a comprehensive comparison of the classification performance across different algorithms and datasets, enabling an in-depth analysis of the results and facilitates drawing robust conclusions.

Dataset	**Model**	**Sentiment Included**				**Sentiment Excluded**			
		Accuracy	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**
Politics	SVM	97.67%	99.28%	95.90%	97.56%	97.68%	99.32%	95.88%	97.57%
	NB	96.71%	94.93%	98.49%	96.68%	96.71%	94.91%	98.51%	96.68%
	DNN	96.45%	98.33%	96.94%	97.63%	96.93%	98.11%	97.19%	97.65%
Health	SVM	84.51%	79.68%	99.59%	88.53%	84.51%	79.68%	99.59%	88.53%
	NB	78.14%	92.31%	69.38%	79.22%	77.71%	93.55%	67.54%	78.45%
	DNN	86.23%	84.91%	97.22%	90.65%	80.98%	83.03%	99.46%	90.51%
Climate Change	SVM	43.97%	39.34%	43.97%	33.12%	44.95%	41.87%	44.95%	34.27%
	NB	35.50%	37.09%	35.50%	36.00%	30.62%	36.83%	30.62%	32.32%
	DNN	44.69%	11.00%	25.00%	15.00%	42.08%	11.00%	25.00%	15.00%
Social Media	SVM	81.34%	35.24%	14.85%	20.89%	81.32%	34.98%	14.65%	20.65%
	NB	53.54%	14.83%	37.93%	21.32%	50.82%	14.65%	40.67%	21.54%
	DNN	82.26%	41.00%	50.00%	45.00%	82.18%	41.00%	50.00%	45.00%

Table 6.1: Overall performance by dataset category and model

It is evident that the SVM classifier outperforms other algorithms, achieving the highest average performance across all dataset categories, with and without the incorporation of sentiment analysis. However, there is an exception in the social media category, where the neural network demonstrates a better performance. In Section 4.4 of the results, specific attention was given to addressing challenges associated with imbalanced data and overfitting. This led to the realization that relying solely on accuracy as a metric can be misleading, as it may appear high due to the dominance of the majority class.

This corresponds to the social media dataset results. Additionally, although the Naive Bayes classifier performed slightly less favorably compared to the other two models, its performance remained competitive and was not significantly different or inferior.

Among the four selected datasets, the politics dataset stands out with the highest performance achieved by the SVM model without incorporating sentiment analysis. When evaluating the performance across different datasets, it is important to consider their structural characteristics, such as size and balance. As demonstrated in Table 3.2 and Figure 3.2 of the previous Chapter, the politics dataset excels in meeting these criteria. It consists of lengthy, less noisy news articles sourced from official news websites (Table 3.2 displays only first two or three sentences of the original sentences of sample records per label). Furthermore, the dataset exhibits a near-balanced distribution, contributing to its superior performance as the best-performing dataset.

Furthermore, when considering the models trained with the incorporation of the sentiment feature, it is evident that the neural network exhibited slightly better performance on imbalanced datasets such as climate change and social media. However, imbalance can often negatively affect neural networks, as highlighted by Ao et al. [aHSSL22]. One potential issue that arises in the context of imbalanced data is overfitting, where the neural network becomes excessively specialized in capturing the patterns of the majority class, resulting in higher accuracy but poor generalization to the minority class. Consequently, this can lead to poor performance in terms of precision, recall, and F1 score, as depicted in Table 6.1.

Finally, when considering the impact of sentiment analysis on enhancing the performance of fake news detection, the comparison Table reveals that sentiment analysis did not lead to significant improvements, and in most cases, resulted in slightly lower performance. One possible explanation for this observation is that fake news and real news exhibited different distributions of sentiment classes, as highlighted in the Quantitative Analysis section. This was evident across all datasets. Previous studies, including Sharma et al. [SQJ$^+$19] and Lin et al. [LKL22], have discussed that each type of news, whether fake or real, tends to exhibit a specific sentiment pattern, such as predominantly negative or positive. However, it was specifically observed that neither fake news nor real news consistently displayed an exaggeration of a specific sentiment type. Consequently, the models struggled to identify consistent patterns to learn from.

6.2 Conclusion

Based on the implications discussed in Chapter 5 and the comprehensive assessments conducted in the preceding Section, this study makes a contribution to the existing body of knowledge in the field of fake news detection approaches by employing diverse datasets from multiple domains and evaluating different classification models. The findings reinforce the suitability of Naive Bayes in resource-constrained scenarios, highlight the effectiveness of SVM with large datasets with lengthy entries such as news articles, and underscore the consistent behavior of Neural Networks and SVM models in similar contexts, with the additional advantage of Neural Networks requiring less computational effort. Thus, the integration of Neural Networks with more balanced data holds the potential to yield improved performance while simultaneously mitigating computational burdens.

CHAPTER 7

Future Research Directions

The methodology employed in this thesis for fake news detection primarily focused on supervised learning methods. While these approaches have provided valuable insights, there are several areas within the field of supervised learning for fake news detection that require further investigation as potential directions for future research. These areas encompass aspects such as enhancing the consistency of multi-source datasets, advancing feature engineering techniques, and refining the analysis process. By expanding and refining the employed methodology, future studies can aim to uncover novel insights that contribute to the effectiveness of fake news detection methods.

7.1 Enhancing Consistency of Multi-Source Datasets

The spread of fake news occurs across various platforms, including online news websites and social media networks. This study demonstrated that the choice of data source can significantly impact the performance of fake news detection models. While social media platforms are known to be a major source of fake news and rumors [MGW18], the short nature of social media posts did not contribute to the improved performance of the applied models, as observed in the results. To address this limitation, Ma et al. [MGW18] suggest extending social media posts to provide additional context. For example, they propose modeling Twitter data as a collection of claims consisting of relevant tweets posted at different times. This approach allows for the establishment of consistency within multi-source datasets from diverse domains and platforms, enabling a more comprehensive analysis of fake news.

7.2 Advancing Feature Engineering

7.2.1 Incorporating Advanced Sentiment-Based Features

As previously discussed in the limitations section, this study primarily incorporates a three-polarity sentiment analysis, considering sentiments as negative, neutral, or positive. However, as demonstrated by Dickerson et al. [DKS14], there is potential for further improvement by exploring various variations of sentiment degree and multi-polarity sentiment analysis. Additionally, incorporating additional sentiment-based features such as average topic sentiment, polarity fractions, positive sentiment strength, and negative sentiment strength can significantly enhance the performance of the classifiers. Table 2.5 in Chapter 2 provides an overview of these features and their relevance to the fake news analysis.

7.2.2 Incorporating Other Content-Based Features

An additional content-based feature that holds potential for enhancing fake news detection is Word2Vec similarity (Word2Vec-Sim). Word2Vec is a technique that represents text data in the Vector Space Model using a two-layer neural network. By building a vocabulary of words and training a model (pre-trained word embeddings such as Google's Word2Vec can be utilized), each word in a sentence can be mapped to an embedding vector. The Word2Vec-Sim feature can then be calculated by computing the cosine similarity between the obtained embedding vectors. By incorporating the Word2Vec-Sim feature, this study aims to capture and measure the semantic similarities between words and sentences. This approach allows for a deeper understanding of textual content and has the potential to enhance the accuracy of classifiers.

7.2.3 Incorporating Context-Based Features

The majority of recent studies have been employing content-based approaches for fake news detection. However, notable studies, such as the work conducted by Castillo et al. [CMP13], have demonstrated the value of incorporating a range of comprehensive context-based features derived from user profiles and news/post characteristics. These features encompass factors such as the popularity level, number of shares.

By considering both content-based and context-based features, and extending the multiple case study by comparing the performance when using all these features to the results when removing features one by one researchers can gain more understanding of their impacts on the classification models. However, it is important to note that not all of these features are available in all types of platforms and therefore bring limitations to a multi-source dataset case study.

Recent studies in the field of fake news detection have mostly focused on content-based approaches. However, notable studies, such as the work conducted by Castillo et al. [CMP13], have demonstrated the value of incorporating a range of comprehensive context-based features derived from user profiles and news/post characteristics. These features encompass factors such as the popularity level, number of shares. By considering both content-based and context-based features, researchers can develop a more comprehensive understanding of the fake news detection problem. To further investigate the impact of these features on classification models, an extended multiple case study can be conducted, comparing the performance when utilizing all features to the results obtained by gradually removing each feature. However, not all platforms provide access to all these features, which may introduce limitations in the case of multi-source dataset studies.

7.3 Refinement of Analysis Degree: Multi-Label Fake News Detection

The majority of existing fake news detection studies, including the focus of this thesis, primarily revolves around binary classification, distinguishing between fake and real news or determining the presence of misleading information. Nevertheless, it is important to acknowledge that text can often comprise a blend of both factual and false statements. Therefore, the exploration of multi-label fake news detection, where a news article can be associated with multiple labels, presents a potential avenue for further investigation. For multi-label fake news detection, where a news article can be associated with multiple labels, there are several suggestions to consider:

- Dataset Expansion: The majority of existing datasets are primarily labeled binary. One suggestion is to expand these datasets by manually annotating them with multiple labels. This can involve leveraging domain experts or crowdsourcing platforms to assign multiple labels to each news article.

- Ensemble Methods: Utilize ensemble methods, such as bagging or boosting, to combine multiple binary classifiers into a multi-label classifier. This approach involves training multiple binary classifiers, each specialized in detecting specific types of fake news, and then combining their predictions to make the final multi-label classification.

CHAPTER 8

Appendix

8.1 Appendix 1: Dataset Sources

Category	Description	Link
Politics	Political News Articles	https://onlineacademiccommunity.uvic.ca/isot/2022/11/27/fake-news-detection-datasets/
Health	COVID-19 Healthcare News and claims	https://github.com/cuilimeng/CoAID
Climate Change	Web-scraped climate change-related claims	https://www.sustainablefinance.uzh.ch/en/research/climate-fever.html
Social Media	Crowdsourced world news-related tweets	https://communitynotes.twitter.com/guide/en/under-the-hood/download-data

Bibliography

[AAQAR$^+$15] Majed AlRubaian, Muhammad Al-Qurishi, Mabrook Al-Rakhami, Sk Md Mizanur Rahman, and Atif Alamri. A multistage credibility analysis model for microblogs. In *Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2015*, pages 1434–1440, Paris, France, 2015.

[AC18] Supanya Aphiwongsophon and Prabhas Chongstitvatana. Detecting fake news with machine learning method. In *2018 15th international conference on electrical engineering/electronics, computer, telecommunications and information technology (ECTI-CON)*, pages 528–531, Chiang Rai, Thailand, 2018. IEEE.

[aHSSL22] Zhan ao Huang, Yongsheng Sang, Yanan Sun, and Jiancheng Lv. A neural network learning algorithm for highly imbalanced data classification. *Information Sciences*, 612:496–513, 2022.

[ARVB16] Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. Stance detection with bidirectional conditional encoding. *arXiv preprint arXiv:1606.05464*, 2016.

[ATS18] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detecting opinion spams and fake news using text classification. *Security and Privacy*, 1(1):e9, 2018.

[AVGRV21] Miguel A Alonso, David Vilares, Carlos Gómez-Rodríguez, and Jesús Vilares. Sentiment analysis for fake news detection. *Electronics*, 10(11):1348, 2021.

[Baj17] Samir Bajaj. The pope has a new baby! *Fake news detection using deep learning*, pages 1–8, Stanford CS224N, 2017.

[Bam18] Noel Bambrick. Support vector machines: A simple explanation. *línea]. Disponible en: https://www. kdnuggets. com/2016/07/support-vector-machines-simple-explanation. html*, 2018.

[BG17] Cody Buntain and Jennifer Golbeck. Automatically identifying fake news in popular twitter threads. In *2017 IEEE international conference on smart cloud (smartCloud)*, pages 208–215, New York, NY, USA, 2017. IEEE.

[BOSB10] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M Buhmann. The balanced accuracy and its posterior distribution. In *2010 20th international conference on pattern recognition*, pages 3121–3124, Istanbul, Turkey, 2010. IEEE.

[CCR] Yimin Chen, Niall J Conroy, and Victoria L Rubin. Misleading online content: recognizing clickbait as" false news". In *Proceedings of the 2015 ACM on workshop on multimodal deception detection*, pages 15–19.

[CL20] Limeng Cui and Dongwon Lee. Coaid: Covid-19 healthcare misinformation dataset. *arXiv preprint arXiv:2006.00885*, 2020.

[CMP13] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Predicting information credibility in time-sensitive social media. *Internet Research*, Vol. 23 No. 5, pp. 560-588, 2013.

[CSR+15] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. Computational fact checking from knowledge networks. *PloS one*, 10(6):e0128193, 2015.

[CTZ20] Matthew Carter, Michail Tsikerdekis, and Sherali Zeadally. Approaches for fake content detection: Strengths and weaknesses to adversarial attacks. *IEEE Internet Computing*, 25(2):73–83, 2020.

[CWB+11] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537, 2011.

[CWY+21] Mingxi Cheng, Songli Wang, Xiaofeng Yan, Tianqi Yang, Wenshuo Wang, Zehao Huang, Xiongye Xiao, Shahin Nazarian, and Paul Bogdan. A covid-19 rumor dataset. *Frontiers in Psychology*, 12:644801, 2021.

[DBGB+20] Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. Climate-fever: A dataset for verification of real-world climate claims. *arXiv preprint arXiv:2012.00614*, 2020.

[DKS14] John P Dickerson, Vadim Kagan, and VS Subrahmanian. Using sentiment to detect bots on twitter: Are humans more opinionated than bots? In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 620–627, Beijing, China, 2014. IEEE.

[DRP+18] Amitabha Dey, Rafsan Zani Rafi, Shahriar Hasan Parash, Sauvik Kundu Arko, and Amitabha Chakrabarty. Fake news pattern recognition using linguistic analysis. In *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, pages 305–309, Kitakyushu, Japan, 2018. IEEE.

[DSBL+] Javier Del Ser, Miren Nekane Bilbao, Ibai Laña, Khan Muhammad, and David Camacho. Efficient fake news detection using bagging ensembles of bidirectional echo state networks. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7.

[GBV20] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*, 2020.

[GLC20] Siva Charan Reddy Gangireddy, Cheng Long, and Tanmoy Chakraborty. Unsupervised fake news detection: A graph-based approach. In *Proceedings of the 31st ACM conference on hypertext and social media*, pages 75–83, Virtual Event, USA, 2020.

[GMBG] Varun Gupta, Rohan Sahai Mathur, Tushar Bansal, and Anjali Goyal. Fake news detection using machine learning. In *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, volume 1, pages 84–89.

[GS15] Marco Guerini and Jacopo Staiano. Deep feelings: A massive cross-lingual study on the relation between emotions and virality. In *Proceedings of the 24th International conference on world wide web*, pages 299–305, Florence, Italy, 2015.

[HA17] Benjamin Horne and Sibel Adali. This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 759–766, 2017.

[HB16] Tanvi Hardeniya and Dilipkumar A Borikar. Dictionary based approach to sentiment analysis-a review. *International Journal of Advanced Engineering, Management and Science*, 2(5):239438, 2016.

[HHK+23] Lee Hadlington, Lydia J Harkin, Daria Kuss, Kristina Newman, and Francesca C Ryding. Perceptions of fake news, misinformation, and disinformation amid the covid-19 pandemic: A qualitative exploration. *Psychology of Popular Media*, 12(1):40, 2023.

[HKS12] Geoffrey E Hinton, Alex Krizhevsky, and Ilya Sutskever. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25(1106-1114):1, 2012.

[JSKG19] Anjali Jain, Avinash Shakya, Harsh Khatter, and Amit Kumar Gupta. A smart system for fake news detection using machine learning. In *2019 International conference on issues and challenges in intelligent computing techniques (ICICT)*, volume 1, pages 1–4, Ghaziabad, India, 2019. IEEE.

[KEP17] Thomas Kautz, Bjoern M Eskofier, and Cristian F Pasluosta. Generic performance measure for multiclass-classifiers. *Pattern Recognition*, 68:111–125, 2017.

[KHRM06] Sang-Bum Kim, Kyoung-Soo Han, Hae-Chang Rim, and Sung Hyon Myaeng. Some effective techniques for naive bayes text classification. *IEEE transactions on knowledge and data engineering*, 18(11):1457–1466, 2006.

[Kie19] Cornelia Kiefer. Quality indicators for text data. *BTW 2019–Workshopband*, pages 145–154, 2019.

[KKA$^+$21] Junaed Younus Khan, Md Tawkat Islam Khondaker, Sadia Afroz, Gias Uddin, and Anindya Iqbal. A benchmark study of machine learning models for online fake news detection. *Machine Learning with Applications*, 4:100032, 2021.

[KKI$^+$19] Junaed Younus Khan, Md Khondaker, Tawkat Islam, Anindya Iqbal, and Sadia Afroz. A benchmark study on machine learning methods for fake news detection. *arXiv preprint arXiv:1905.04749*, 2, 2019.

[KKK20] Sawinder Kaur, Parteek Kumar, and Ponnurangam Kumaraguru. Automating fake news detection system using multi-level voting model. *Soft Computing*, 24(12):9049–9069, 2020.

[KRSST18] Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. Multi-source multi-class fake news detection. In *Proceedings of the 27th international conference on computational linguistics*, pages 1546–1557, Santa Fe, New Mexico, USA, 2018.

[KVV22] Jacqueline Kazmaier and Jan H Van Vuuren. The power of ensemble learning in sentiment analysis. *Expert Systems with Applications*, 187:115819, 2022.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[LKL22] Szu-Yin Lin, Yun-Ching Kung, and Fang-Yie Leu. Predictive intelligence in harmful news identification by bert-based ensemble learning model with text sentiment analysis. *Information Processing & Management*, 59(2):102872, 2022.

[LW18] Yang Liu and Yi-Fang Wu. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, 32(1), 2018.

[MDP+11] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[MGM15] Yajie Miao, Mohammad Gowayyed, and Florian Metze. Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 167–174, Scottsdale, AZ, USA, 2015. IEEE.

[MGW18] Jing Ma, Wei Gao, and Kam-Fai Wong. Detect rumor and stance jointly by neural multi-task learning. In *Companion proceedings of the the web conference 2018*, pages 585–593, Lyon, France, 2018.

[Moh16] Saif M Mohammad. Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In *Emotion measurement*, pages 201–237. Elsevier, 2016.

[MRD+22] Paul Meddeb, Stefan Ruseti, Mihai Dascalu, Simina-Maria Terian, and Sebastien Travadel. Counteracting french fake news on climate change using language models. *Sustainability*, 14(18):11724, 2022.

[MSS22] Mohit Mayank, Shakshi Sharma, and Rajesh Sharma. Deap-faked: Knowledge graph based approach for fake news detection. In *2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 47–51, Istanbul, Turkey, 2022. IEEE.

[PU+19] Karishnu Poddar, KS Umadevi, et al. Comparison of various machine learning models for accurate detection of fake news. In *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, volume 1, pages 1–5, Vellore, India, 2019. IEEE.

[QRRM11] Vahed Qazvinian, Emily Rosengren, Dragomir Radev, and Qiaozhu Mei. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 1589–1599, 2011.

[RL15] Victoria L Rubin and Tatiana Lukoianova. Truth and deception at the rhetorical structure level. *Journal of the Association for Information Science and Technology*, 66(5):905–917, 2015.

[SCV+18] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. The spread of low-credibility content by social bots. *Nature communications*, 9(1):1–9, 2018.

[Seb02] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[SMW+20] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big data*, 8(3):171–188, 2020.

[SQJ+19] Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3):1–42, 2019.

[SSW+17] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36, 2017.

[Sta17] Kate Starbird. Examining the alternative media ecosystem through the production of alternative narratives of mass shooting events on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 230–239, 2017.

[SZ17] Sadia Sharmin and Zakia Zaman. Spam detection in social media employing machine learning tool for text mining. In *2017 13th international conference on signal-image technology & internet-based systems (SITIS)*, pages 137–142, Jaipur, India, 2017. IEEE.

[TAR16] Abinash Tripathy, Ankit Agrawal, and Santanu Kumar Rath. Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, 57:117–126, 2016.

[TAS+20] Jafar Tanha, Yousef Abdi, Negin Samadi, Nazila Razzaghi, and Mohammad Asadpour. Boosting methods for multi-class imbalanced data classification: an experimental review. *Journal of Big Data*, 7(1):1–47, 2020.

[VPS18] Krishna B Vamshi, Ajeet Kumar Pandey, and Kumar AP Siva. Topic model based opinion mining and sentiment analysis. In *2018 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–4, Coimbatore, India, 2018. IEEE.

[VST+09] June M Verner, Jennifer Sampson, Vladimir Tosic, NA Abu Bakar, and Barbara A Kitchenham. Guidelines for industrially-based multiple case studies in software engineering. In *2009 Third International Conference on Research Challenges in Information Science*, pages 313–324, Fez, Morocco, 2009. IEEE.

[VSVR16] Prashanth Vijayaraghavan, Ivan Sysoev, Soroush Vosoughi, and Deb Roy. Deepstance at semeval-2016 task 6: Detecting stance in tweets using character and word-level cnns. *arXiv preprint arXiv:1606.05694*, 2016.

[WAL+14] You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. Toward computational fact-checking. *Proceedings of the VLDB Endowment*, 7(7):589–600, 2014.

[Wan17] William Yang Wang. " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*, 2017.

[WD17] Claire Wardle and Hossein Derakhshan. *Information disorder: Toward an interdisciplinary framework for research and policymaking*, volume 27. Council of Europe Strasbourg, 2017.

[WWW12] Liwei Wei, Bo Wei, and Bin Wang. Text classification using support vector machine with mixture of kernel. *Journal of Software Engineering and Applications*, 5:55, 2012.

[YSW+19] Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu. Unsupervised fake news detection on social media: A generative approach. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5644–5651, 2019.

[ZM20] Melissa Zimdars and Kembrew McLeod. *Fake news: understanding media and misinformation in the digital age*. MIT Press, 2020.

[ZZX17] Qi Zeng, Quan Zhou, and Shanshan Xu. Neural stance detectors for fake news challenge. *CS224n: natural language processing with deep learning*, 2017.