IMT School for Advanced Studies, Lucca
Lucca, Italy

Discovery of False Information on Digital Communication Channels

PhD Program in Cybersicurezza
Track in Software, System, and Infrastructure Security
XXXVIII Cycle

By

Farwa Batool

2026

The dissertation of Farwa Batool is approved.

PhD Program Coordinator: Prof. Mirco Tribastone, IMT School for Advanced Studies Lucca

Advisor: Prof. Giuseppe Lo Re, University of Palermo

Co-Advisor: Prof. Marco Morana, University of Palermo

The dissertation of Farwa Batool has been reviewed by:

Prof. Salvatore Monteleone, Niccolò Cusano University

Prof. Carlo Vallati, University of Pisa

IMT School for Advanced Studies Lucca
2026

Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my deepest gratitude to my advisor, Prof. Giuseppe Lo Re, and my co-advisor, Prof. Marco Morana, for their invaluable guidance, support, and feedback throughout my research period. Their encouragement and mentorship have been fundamental to the completion of this work.

I am also sincerely thankful to my colleagues at NDSLAB, especially Federico, Salvatore, and Marena, for their constant support, collaboration, and for sharing both the challenges and the memorable moments along this journey.

Above all, I owe my deepest appreciation to my parents, sisters, and fiancé, whose unconditional love, encouragement, and faith in me have been a source of strength and motivation during the most crucial moments of my life.

Finally, I acknowledge that the AI-based tools were only used for grammar correction and clarity of sentence structure, while all scientific contributions remain entirely mine and of my co-authors.

Vita

April 01, 1997	Born, Layyah, Pakistan
2014-2018	Bachelors in Computer Engineer Final mark: 3.25/4.00 UET, Taxila, Pakistan
2018-2021	Masters in Computer Engineer Final mark: 3.763/4.00 UET, Lahore, Pakistan
2019-2021	Graduate Assistant University of Engineering and Technology (UET) New Campus, Lahore, Pakistan
2021-2022	Teaching Fellow University of Engineering and Technology (UET) Lahore, Pakistan
2022-2023	Lecturer University of Engineering and Technology (UET) New Campus, Lahore, Pakistan
2023-Present	Ph.D. Scholar IMT Alti Studi Lucca Lucca, Italy

Publications

1. F. Batool, F. Canino, F. Concone, G. Lo Re, and M. Morana, "A Black-box Adversarial Attack on Fake News Detection Systems" *Italian Conference on Cyber Security (ITASEC 2024)*.
2. F. Batool, G. Lo Re, and M. Morana, "Annotated Dataset Creation for Fake News Detection on Online Social Networks" *International Conference on Advanced Information Networking and Applications*, pp. 48-58, 2025. Cham: Springer Nature Switzerland, doi:10.1007/978-3-031-87778-0_5
3. F. Batool, G. Lo Re, M. Morana and M. Tortorici, "Multi-Domain Fake News Detection exploiting Ensemble Learning Techniques" *CEUR Workshop Proceedings, Joint National Conference on Cybersecurity, ITASEC & SERICS* 2025.
4. F. Batool, G. Lo Re, M. Morana and G. Rizzo, "Human Activity Recognition Through Probabilistic Data Fusion," 2025 IEEE *International Conference on Smart Computing (SMARTCOMP)*, Cork, Ireland, 2025, pp. 438-443, doi: 10.1109/SMARTCOMP65954.2025.00100.
5. V. Agate, F. Batool, A. Bordonaro, A. De Paola, P. Ferraro, G. Lo Re, et al. (2025). "Population Protocols for Adaptive Event Dissemination with Autonomous Agents in Vehicular Networks" *17th International Conference on Agents and Artificial Intelligence* pp. 640-647, doi: 10.5220/0013341100003890.
6. F. Batool, G. Lo. Re, M. Morana and G. Rizzo, "Blocking Fake News Propagation exploiting OSNs Users Interests and Connections," *2025 5th Intelligent Cybersecurity Conference (ICSC)*, Tampa, FL, USA, 2025, pp. 49-54, doi: 10.1109/ICSC65596.2025.11139926.
7. F. Batool, G. Lo. Re, M. Morana and U. Ghani, "Past to Present-the Evolution of Fake News Detection Techniques" *2025 Computers, Communications and IT Applications Conference*, Madrid, Spain, {IN PRESS}
8. A. Azam, F. Batool, G. Lo. Re, M. Morana and U. Ghani, "Generative-AI vs. Traditional Methods for Fake News Detection" *2025 4th International Conference on Computing, Management and Telecommunications (ComManTel)*, Madrid, Spain, {IN PRESS}

Presentations

1. F. Batool, "On the Black-box Adversarial Attack on Fake News Detection Systems", at *Italian Conference on Cyber Security*, Salerno, Italy, 2024.
2. F. Batool, "On Annotated Dataset Creation for Fake News Detection on Online Social Networks" at *International Conference on Advanced Information Networking and Applications*, Switzerland, 2025
3. F. Batool, "On Multi-Domain Fake News Detection exploiting Ensemble Learning Techniques" at *CEUR Workshop Proceedings, Joint National Conference on Cybersecurity, ITASEC & SERICS*, Bologna, Italy, 2025.
4. F. Batool, "On Past to Present-the Evolution of Fake News Detection Techniques" at *2025 Computers, Communications and IT Applications Conference*, Madrid, Spain, 2025.

Abstract

From rumors in ancient marketplaces to propaganda in print media, misinformation has always been part of human societies. What has changed in the modern era is the scale, speed, and reach with which false information spreads. The major contribution in this spread is the accessibility and immediacy of modern communication channels. Unauthorized content is uploaded and widely broadcast, often being absorbed and believed by unsuspecting users. This issue jeopardizes public trust, influences their perceptions, and in severe cases, endangers their health.

Robust automatic systems are essential to address this problem. Therefore, a wide range of advanced methods is being explored by researchers for fake news detection on online social networks. However, despite the considerable progress achieved, several limitations still exist. These challenges primarily relate to the inadequacy of updated datasets, the lack of integration of contextual information, the insufficient approaches for detecting and controlling fake news on social networks, and the absence of evaluation of system resilience.

This thesis aims to bridge these gaps by providing effective and practical solutions. First, a comprehensive and up-to-date dataset for fake news detection is presented, enriched with detailed information to support further research. Second, an efficient system for fake news detection is proposed, utilizing the strength of ensemble learning while integrating domain knowledge of news. Third, an innovative approach is proposed to limit the dissemination of fake news on online social networks. Lastly, it analyzes the strength of existing

fake news detection models through a novel adversarial attack architecture.

Chapter 1

Introduction

In today's digital world, individuals are constantly surrounded by an overwhelming amount of information, accessed through traditional news outlets, digital media channels, and online communication platforms. Human decision-making often relies on quick judgments, emotional reactions, and trusted sources. However, when these natural tendencies meet the vast and unfiltered content online, the risk of misinterpreting or accepting false information increases dramatically.

A notable example emerged during the COVID-19 pandemic, when the novelty of the disease, the absence of reliable treatment, and the intensity of fear prompted many individuals to seek guidance through online platforms such as Twitter, YouTube, Facebook, and Instagram. In this context, numerous non-professionals circulated unverified home remedies for prevention or treatment against the virus, and many individuals adopted these practices (Rocha et al., 2023). While some remedies were benign, others posed significant health risks, illustrating the dangers associated with the dissemination and adoption of misinformation in critical cases.

Misinformation, also referred to as fake news, is not only harmful in the context of health and medicine, but it also affects areas such as politics, education, and entertainment. The reason behind the creation and spread of fake news can be as simple as a joke or as harmful as manip-

ulating the public's opinion against a certain topic, individual, or community. Therefore, there is a pressing need for early detection of false information on online social networks and working towards its blockage before it spreads.

To address this challenge, research efforts progressively shifted towards developing optimal solutions for the timely detection and containment of misinformation. Initially, the process relied on manual labeling of news articles suspected to be fake, carried out by domain experts. While this approach ensured accuracy, it required a considerable amount of time and resources, making it unsuitable for large-scale applications. To automate this process, machine learning models were introduced, achieving good results by identifying patterns in fake news. With the advancement of research, more sophisticated architectures, such as deep learning models, have been deployed, offering improved performance. More recently, large language models (LLMs) have been employed not only for the detection of fake news (Teo et al., 2024) but also for its generation (Papageorgiou, Chronis, et al., 2024), allowing researchers to assess the robustness and effectiveness of detection systems under increasingly realistic conditions.

Researchers try to find out the causes of the generation and spread of fake news. For this purpose, they focus on features such as identifying and blocking malicious users or accounts. To analyze the patterns of propagation of false news, the network structures and propagation patterns are studied in detail. Additionally, other types of research focus on the identification of text, images, or videos that spread fake narratives.

1.1 Motivation and Goals

Although a significant amount of research is being done on the detection and containment of false news, some limitations exist in this domain. This thesis aims to address several gaps by providing efficient and robust solutions. The currently available benchmark datasets for fake news detection are not only old but also lack details related to the news. Having the most information about the news can help the models identify pat-

terns easily, resulting in better performance. Secondly, for the detection of fake news, an important feature is the integration of information regarding the domain through which the news belongs. The writing styles and characteristics of the text differ between domains, which makes it easier for systems to classify fake news easily when the domain information is given. Nevertheless, the literature lacks an efficient system incorporating domain information for fake news detection. Additionally, fake news on online social platforms needs to be detected quickly, and its spread must be controlled. Detection and containment rely on various features, and identifying them is crucial, yet often overlooked in the literature. Finally, an extensive analysis of literature highlighted that traditional machine learning models, despite their simplicity and promising detection abilities, are vulnerable to adversarial attacks. However, the existing attacking mechanisms are time-consuming and prone to being identified by the target system.

This thesis addresses the aforementioned challenges and proposes effective solutions, contributing to multiple key dimensions of fake news detection. Specifically, it advances the field through adversarial machine learning for robustness evaluation, content-based detection approaches, the creation of a comprehensive dataset, and propagation-based fake news detection.

1.2 Dissertation Outline

This thesis is a compilation of work derived from the author's previously published articles, which are listed in the publication section and appropriately cited in the corresponding chapters. The remainder of this dissertation follows this structure.

Chapter 2 provides the essential background to understand the work done in fake news detection. Moreover, it discusses features and domains of fake news based on which the models are trained and tested. Lastly, the gaps in the literature are discussed, which will be addressed in this thesis.

In Chapter 3, an architecture is proposed to create a holistic dataset

for fake news detection. Chapter 4 discusses an ensemble architecture that effectively incorporates domain knowledge of fake news. Chapter 5 presents a novel approach to limit the propagation of fake news on online social networks. In Chapter 6, an efficient method for adversarial attack is proposed for a black-box fake news detection system. Chapter 7 concludes this thesis.

Chapter 2

Literature Review

In an era where misinformation propagates through digital platforms, effective fake news detection is of paramount importance. This literature survey provides a comprehensive overview of features, methods, and techniques of fake news detection. Lastly, the existing limitations are explored and discussed.

2.1 Existing models

To combat the rapidly spreading misinformation across platforms, initial measures were taken, such as fact-checking websites[1][2][3][4][5]. These websites are accessible for evaluating the truthfulness of specific claims or news stories. They operate manually, with domain experts assessing the accuracy based on their experience and connections. Multiple experts review a piece of news, and the aggregated label is assigned. Although effective, this method is quite time-consuming and requires manual effort. Therefore, researchers shifted their focus toward automatic and timely detection of fake news. To achieve this goal, artificial intelligence proved

[1] FactCheck: https://www.factchecker.in
[2] Politifact: https://www.politifact.com
[3] Suggest (formerly: Gossipcop): https://www.suggest.com
[4] Snopes: https://www.snopes.com
[5] Washington Post Checker: https://www.washingtonpost.com/news/fact-checker

useful.

2.1.1 Machine Learning Models

In the domain of machine learning, a variety of models are widely employed for classification and prediction tasks. Support Vector Machines (SVMs) (Baarir and Djeffal, 2021), for instance, are discriminative models that identify an optimal hyperplane to separate data points into distinct categories. Decision Trees (Lyu and Lo, 2020) operate by recursively partitioning the feature space, forming a tree-like structure that enables interpretable decision-making. Similarly, the K-Nearest Neighbor (KNN) algorithm (Gravanis et al., 2019) assigns a data point to the class most common among its closest neighbors. Naïve Bayes (Khanam et al., 2021), grounded in Bayesian probability, is particularly effective for classification problems, especially in text-related tasks. In contrast, Linear regression (Vijayaraghavan et al., 2020) is typically applied to regression problems, where the goal is to predict continuous numerical outcomes. Although these models achieved promising results in the news text classification, they were limited by their inability to capture the deeper contextual meaning of the content. This highlighted the need for more advanced approaches capable of going beyond surface-level pattern recognition to incorporate contextual understanding in their prediction.

2.1.2 Deep Learning Models

To overcome the problem of a lack of contextual learning, deep learning models contributed significantly by leveraging neural networks. These models learn and often remember the contextual relationships between text. The most frequently utilized deep learning models encompass the Recurrent Neural Network (RNN) (Z. Jin et al., 2017), an architecture of neural networks adept at processing sequential data, employed to encapsulate temporal dependencies in news articles. While Long Short-Term Memory (LSTM) (Bahad, Saxena, and Kamal, 2019) is an RNN variant capable of capturing long-range dependencies in sequential data, making it suitable for analyzing textual information over extended contexts.

Similar to LSTM, GRU is employed in RNNs due to its ability to model sequential information efficiently, particularly in scenarios where computational resources are constrained (Buzea, Trausan-Matu, and Rebedea, 2022). Initially designed for image analysis, CNNs are also adapted to process text and extract features from textual content (Karimi et al., 2018). These models, despite their increased complexity, demonstrated greater efficiency than traditional machine learning approaches, particularly in capturing contextual information and providing more accurate predictions.

2.1.3 Large Language Models

While deep learning models significantly improved fake news detection compared to traditional machine learning models, they still exhibited significant limitations. Firstly, they required large amounts of labeled data, making their performance highly dependent on data availability and quality. Secondly, the basis of their prediction remained unexplained. In these cases, Large Language Models (LLMs) proved to be much effective. LLMs are pre-trained on large corpora and can be fine-tuned to generalize well on any problem, even with limited labeled data. Furthermore, explainable AI has enhanced the interpretability of LLMs, offering greater transparency in decision-making compared to conventional deep learning approaches. The models currently being employed in the literature include BERT (Papageorgiou, Varlamis, and Chronis, 2025), OpenAI's ChatGPT 3.5 (M. Chen et al., 2023) GPT-4, Claude 3 Sonnet, Gemini Pro 1.0, and Mistral Large (Koka, Vuong, and Kataria, 2024). LLMs are delivering impressive results in fake news detection, but it remains an under-explored area.

2.1.4 Ensemble Learning

In addition to the models mentioned above, ensemble techniques are often deployed to enhance the predictive power of fake news detection models. These techniques involve combining multiple models to achieve improved performance. Ensemble methods can encompass various strate-

gies, such as bagging, boosting, and stacking, and are tailored to leverage the strengths of individual models while mitigating their weaknesses (Tufchi, Yadav, and Ahmed, 2023). Ensemble techniques further enhance the predictive accuracy of these models, collectively contributing to the development of robust and effective solutions for addressing the pervasive issue of fake news.

2.2 Features

Since model selection is a crucial step in detection and prediction, another key step is feature extraction and selection. Features serve as the fundamental building blocks of data, enabling the recognition of patterns that models can learn from to perform effectively on new, unseen data. In fake news detection, the feature space encompasses News Content-based features, Social-context based features, and User- based features.

2.2.1 News Content-Based Features

Features with a primary focus on different aspects of news articles, such as their content, style, and source, are classified as news content-based features. These features are evaluated at the word, content, or sentence level Kondamudi et al., 2023. The fake news is written in a way to attract the readers. A careful analysis of the creator's language, word choice, and writing style can provide valuable cues for distinguishing between fake and real news. The images portrayed in fake news are more aggressive and emotional than those in real news. Additionally, the quality of fake news often differs significantly from that of real news. Additionally, image tempering and manipulation detection can also lead to better detection. Verma et al. (2021) proposed a two-level model, named WELFake, based on three sets of linguistic features. These feature sets, comprising 87 linguistic attributes, were integrated with word embeddings and evaluated using a voting-based classification approach. The results showed that WELFake outperformed the traditional fake news detection models.

2.2.2 Social Context-Based Features

Context-based fake news detection scrutinizes the propagation dynamics of news. Features encompass temporal aspects, network structures, impact, and distribution patterns, allowing for an in-depth understanding of when, how, and which information spreads (X. Zhang and Ghorbani, 2020).

Internet news publishers often employ a *temporal-based approach* represented in a time-series format. This strategy allows the analysis of temporal features, which can be instrumental in assessing the authenticity of social media information. By examining the timing and sequencing of content publication, temporal features offer valuable insights into detecting potentially fraudulent publishing activities (Y. Guo and Song, 2022).

An effective approach for gauging the veracity of online news is through the examination of the *network structure* surrounding the news creator (Kempe, Kleinberg, and Tardos, 2003). This entails analyzing the connections, relationships, and interactions within the network to discern patterns indicative of truthful or deceptive content dissemination.

While the network analysis focuses on understanding the interactions among online users, distribution pattern analysis concentrates on unraveling the characteristics of information propagation (X. Zhou and Zafarani, 2019). It involves the study of how news spreads, identifying distinctive patterns and behaviors that can aid in differentiating between genuine and deceptive information dissemination.

2.2.3 User-Based Features

User-based fake news detection, on the other hand, revolves around assessing the credibility of the individuals sharing or posting news. This category delves into user profiles, credibility metrics, and user behavioral patterns, focusing on human element in the dissemination of information.

- **User Profiling:** User profiling involves gathering comprehensive information about the user involved in the dissemination of news,

whether they are a news publisher or a content spreader. This category of features includes details such as the user's name, account information, and the nature of their online activities, providing insights into their online presence.

- **User Credibility Assessment:** This category delves into evaluating the credibility of the user sharing or publishing news. It includes metrics such as the total number of posts and comments made by the user, the frequency of their posting, the size of their followers and friends network, and other relevant indicators. Assessing user credibility is vital in discerning the reliability of news sources.

- **User Behavior Analysis:** User behavior features offer understanding of how individuals interact with news content. This type of feature extraction encompasses the examination of the user's posting habits, the types of posts they engage with, their commenting patterns, and other behavioral cues. Analyzing user behavior is instrumental in identifying any patterns associated with the dissemination of fake news.

These diverse feature categories collectively empower models to effectively discern fake news in the complex landscape of information dissemination.

2.3 Domains

In addition to the features mentioned above, domain knowledge plays a crucial role in detecting fake news. In this context, "domain" refers to the area or category to which the news belongs. It could be related to politics, entertainment, health, education, or any other sector of life. Integrating the information related to the domain to which the news belongs allows better generalization of classifiers.

2.3.1 Single Domain Fake News Detection

The researchers have keenly observed and experimented on various domains of fake news. One of the methods adopted by Agarwal et al. (2019) utilizes five classifiers on a political database, leveraging bag-of-words, n-grams, count vectorizer, and TF-IDF. Raza (2021) also proposed a Transformer-based architecture to predict the probability of a news to be real or fake. They utilized a dataset consisting of news from the 2020 U.S. election. The experiments provided promising results with higher accuracy compared to the baseline models. Konkobo et al. (2020) utilized the dataset *FakeNewsNet*, which is a combination of news from entertainment and political domains. They have utilized a CNN-based approach to find out that the proposed models achieved more than 70% accuracy on both domains.

Although misinformation affects every area of life, the greatest danger is caused in the health domain. People believe what they see on social media and apply to themselves, resulting in even worse conditions (Loeb et al., 2020; Barua et al., 2020). Therefore, researchers have explored this area with a great focus. Mahara et al. (2023) proposed a comparative study between deep learning and machine learning models for health-related fake news detection. While Shrivastava and Sharma (2022) and Barua et al. (2020) introduced a pre-tuned Bert-based model for COVID-19-related misinformation spreading on social media.

2.3.2 Multi-Domain Fake News Detection

Fake news can spread widely across topics and platforms. A news piece cannot be declared as from one domain only. Moreover, the news content style and propagation pattern also differ across the platforms. That being the reason to shift towards multi-domain detection. To address the complexities of domain shift and domain label incompleteness, a multi-faceted approach was proposed by Zhu et al. (2022) for fake news detection. This innovative model, devised to address the challenge of multi-view analysis, captures semantic, emotional, and stylistic information from the news articles. Panda and Levitan (2022), on the other hand, per-

formed cross-domain deception detection on five domains, through distance metrics. Another interesting research (Silva et al., 2021) addresses this challenge by proposing a novel framework that preserves domain-specific and cross-domain knowledge. Additionally, it introduces an unsupervised technique to choose informative unlabeled news records, for manual labeling, from a large pool of unlabeled records. Thus, it reduces labeling costs. Nan et al. (2021) introduced a straightforward yet highly efficient technique referred to as Multi-domain Fake News Detection (MDFEND), and evaluated its performance on a multi-domain dataset collected from Chinese websites. While the multi-domain fake news detection is of prime importance, Daokang Wang et al. (2023) emphasized that multi-domain labels are critical, as the absence of explicit domain labeling hinders the acquisition of domain-specific knowledge. Consequently, models trained without such labels may produce unreliable results. Thus, a multi-domain fake news detection model, SLFEND, was proposed. Y. Sun et al. (2024) proposed a memory guidance-based multi-domain fake news detection model (MDFM), focusing on the relationship between publisher and social emotions. Goel et al. (2021) utilized transfer learning using attention-based transformers across multi-domain datasets to achieve an accuracy of 99%. Additionally, multiple research studies are being conducted on this aspect, including visual information Qi et al. (2019), User Engagement X. Yang et al. (2024), and History News W. Yu et al. (2022), among others.

2.4 Models' Vulnerability

Although extensive research is being conducted on the development of efficient models for detecting fake news on social media platforms, one critical aspect often overlooked is the strength and security of these models. Z. Zhou et al. (2019) demonstrated that the models based on linguistic features are highly vulnerable to tempering attacks. Similarly, Koenders et al. (2021) validated this concern by applying text-based adversarial attacks on widely used fake news detection systems. On the other hand, the approach employed by Rath, W. Gao, and Srivastava (2019)

emphasized that beyond model vulnerability, the trust relationships between nodes (users) within a network significantly influence the spread and believability of fake news. Recently, L. Wang et al. (2024) proposed a black-box attack on user engagement while preventing the fake news from being detected by automated systems.

2.5 Gaps

After the thorough review of the literature, several key limitations have been identified and are discussed comprehensively in further chapters. A critical gap in the literature is the absence of a comprehensive and up-to-date dataset for fake news detection. Additionally, while certain models have been created to address multi-domain news, they still lack effectiveness and robustness. Furthermore, the timely identification of false content on social networks and the implementation of effective measures to prevent its spread are essential. Ultimately, our analysis demonstrates that the existing fake news detection models remain vulnerable to adversarial attacks. However, upon closer examination, it was observed that the adversarial attacks are not always practically feasible, as they often introduce challenges for the attacker.

Therefore, the following are problem statements for this thesis;

- The scarcity of comprehensive and up-to-date annotated datasets for fake news detection, hindering the development and benchmarking of novel approaches.
- The lack of effective frameworks capable of handling multi-domain fake news detection with robustness.
- The need to integrate factors influencing the spread of misinformation during fake news detection and simultaneously developing measures to control its propagation.
- The absence of efficient adversarial attack methods on fake news detection models, as existing approaches are computationally ex-

pensive and easily detectable, limiting their practicality for robust evaluation.

This thesis aims to address the above-mentioned gaps by proposing effective and well-founded solutions.

Chapter 3

Annotated Dataset Creation for Fake News Detection

In the literature, several methods exist that allow detecting fake news in circulation, including: traditional machine learning methods (Gravanis et al., 2019), deep learning methods (Raza and Ding, 2022), knowledge-based detection methods (Seddari et al., 2022), propagation-based detection methods (Meyers, Weiss, and Spanakis, 2020), and source-based detection methods (Sitaula et al., 2020), which consider some linguistic, temporal, user information-based, and interaction-based characteristics. These models exhibit a strong dependency on data, necessitating large-scale and diverse datasets to capture domain-specific nuances and complexities. This data intensity is particularly crucial to achieve robust predictive accuracy and generalizability. The type of information collected from the datasets depends on the application's purpose and may vary significantly between datasets. For example, some datasets focus on rumor-based news (Zubiaga et al., 2016) while others include political (Santia and Williams, 2018; Shu, Mahudeswaran, et al., 2020) or health-related statements (Du et al., 2021; Patwa et al., 2021). Furthermore, they also differ according to the type of content included (e.g., user responses, source of the statement) and the labels that are provided. Datasets are often used as training or validation models. This means that the number of

labels in the dataset, quantity, and quality of data influence the classification algorithms for fake news detection. These considerations highlight, on one hand, the significant influence of datasets on fake news detection algorithms, and on the other hand, the limitation in the quality of the available information. This underscores the need for a large-scale system to collect tweets. Therefore, a system that gathers most information about tweets and users is presented. Then, the collected dataset is utilized to test unsupervised algorithms for fake news detection, which are known in the literature, to analyze the validity of the proposed system.

3.1 Methodology

The problems related to current fake news detection methods and the need for annotated datasets have led to the experimentation of a system for mass collection of tweets with semi-automatic annotation. As mentioned in (Batool, Lo Re, and Morana, 2025), the proposed system consists of two phases: one for collecting tweets with complete related information and the other that attributes a truth value to the collected tweets. The goal of the system is to create a dataset containing tweets and all the necessary information associated with them, intending to help the research community by providing a tool to create annotated datasets and help define new techniques. To evaluate the performance of the algorithms, it is necessary to establish a procedure that allows for determining a Ground Truth value and establishing whether each tweet in the dataset is fake or real.

A modular architecture characterizes the system, and therefore includes a sequence of phases as outlined in Figure 1.

The process begins with the input of a set of keywords related to certain topics, such as politics, news, health, and gossip. The X platform (previously Twitter) offers APIs that enable users to retrieve specific tweets straightforwardly and uniquely. The API provides access to a set of methods that allow you to obtain a large amount of information about tweets. The system responds with all related attributes, including tweet text, tweet ID, creation date, embedded URLs or mentions, number

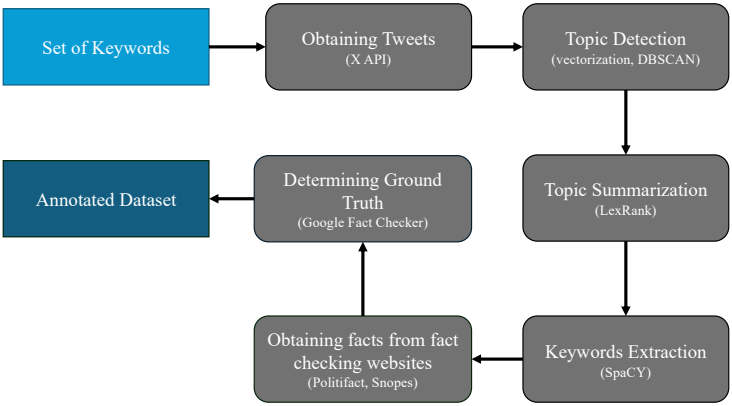


Figure 1: System Overview

of likes, number of retweets, media attached to the tweet, hashtags, and several additional metadata features. There is also information about the user who published the tweet, including the user's name, user ID, account creation date, number of followers, number of friends, and user timeline. Tweets can be collected either by their identifier or by a word present in the tweet. In this case, tweets were obtained using a set of keywords, both because tweet IDs are not available, and tweets related to a specific topic can be collected. So, for each keyword present in the input set, a request must be made to the X platform via the relevant API method. The Python library *tweepy* was used for data collection, which allows access to the X API.

The obtained collection constitutes input of the *topic detection* phase, which aims to group tweets by clustering semantically similar ones rep-

resenting the same claim.

Clustering requires vectorizing tweets into numerical form, which is done using a binary matrix where rows represent tweets and columns represent words. The presence of a word in a tweet is marked with 1 using the CountVectorizer. Then, the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) (Ester et al., 1996) algorithm is used. DBSCAN connects points based on density, identifying regions with similar densities and isolating outliers. It does not require a predefined number of clusters and can detect clusters of arbitrary shapes. The algorithm needs two parameters: a radius R and a minimum number N of points, which are determined by evaluating clustering performance through the Silhouette coefficient (Steenari and Nurminen, 2023). The highest Silhouette coefficient, which ranges from -1 to 1, is used to select the best parameters, with higher values indicating better clustering.

$$S_i = \frac{Dmin_i^{out} - Davg_i^{in}}{max(Dmin_i^{out}, Davg_i^{in})}$$

Here, $Davg_i^{in}$ indicates the average distance of a point of a cluster from the points of its own cluster, while $Dmin_i^{out}$ represents the minimum average distance of a point from other clusters different from its own. The clustering thus provides the output containing: CLAIM, which shows the cluster identifier; NUM_OF_TWEETS, which represents the number of tweets that make up the cluster; LIST_OF_ID, which includes a list of IDs assigned to the clusters; and LIST_OF_TEXT, which consists of a list of the texts of the tweets assigned to the cluster. This output is forwarded to the next component for the topic summarization.

After clustering the tweets, the subsequent step involves identifying and formulating the representative claim associated with each cluster. For this purpose, the **LexRank** approach is used, which is based on the fact that a sentence similar to many other sentences in the text has a high probability of being important. The more frequent the sentence, the higher the rank, which constitutes the priority of being included in the summarized text. This approach is suitable for the purpose, since the most frequent sentences in the tweets are most significant for the fact represented. LexRank summarizes the text and pro-

vides the following attributes: CLAIM representing the claim identifier; LIST_OF_ID, which includes a list of tweet IDs assigned to the clusters; LIST_OF_TEXT, which is a list of the texts assigned to the clusters; and LEX_RANK_SUMMARY_TEXT consists of the summary obtained from the LexRank model.

The next phase regards obtaining *facts*. To this aim, Google Fact Check Tools is used to perform queries by one or more keywords, through which you can obtain facts annotated by the most popular fact-checking sites. Since the search must be through keywords, a process is needed that, starting from the text of the claim determined by summarization, allows you to obtain a set of keywords. When processing input, requiring all words to be present in the facts reduces the likelihood of finding relevant information, as each fact must include all input words. This phase aims to remove non-essential words from the claim texts while maintaining their original meaning. Keywords' extraction, significant from the claims, was performed through the spaCy library, which returns INITIAL_KEYWORDS, which are the tokens resulting from the grammatical filtering, and DEFINITIVE_KEYWORDS constituting the list of words obtained from the second and final filtering. These two attributes, along with the CLAIM and LEX_RANK_SUMMARY_TEXT from the previous step, are forwarded to the next step. After obtaining significant keywords for each claim, a request is made to the Google Fact Check system specifying the set of keywords and Politifact.com and Snopes.com as fact-checking sites. The response obtained from the system is a set of annotated facts. For each fact-checking site considered, it is necessary to transform the ratings of the relative site into the two classic truth values: *True* or *False*. To assign a single truth value to a claim in the dataset, the following method is used: if only one fact is retrieved, its truth value is assigned to the claim. When multiple facts are obtained, the truth value of the fact most similar to the claim is used. Texts of all facts and the claim are vectorized using sklearn's CountVectorizer, converting them into binary vectors indicating word presence. The claim vector is then compared to fact vectors using Euclidean distance, and the truth value of the closest fact is assigned to the claim. Finally, the system-generated

annotation is manually verified to ensure the accuracy of assigned truth values. If the fact corresponding to the claim aligns in terms of information content and meaning, the truth value associated with the fact is assigned to the claim. Conversely, the opposite truth value is assigned if the fact conveys the opposite meaning or denies the claim. In cases where the statements of the fact are unrelated or neutral (not attributable to either True or False), the claim in question is assigned the attribute *Undetermined*. Tweets with Undetermined or Unknown values are discarded, leaving only True and False tweets to evaluate the fake news detection algorithms. Additionally, a dataset of users' followers was collected to enable analysis of user relationships. The tweepy Python library's *api.friends()* method was used to gather the 20 most recent followers for each user.

3.2 Feature Analysis and Detection Techniques

To evaluate the proposed methodology, multiple established methods from the literature were selected, namely EM (Dong Wang et al., 2012), EM-MultiF (Shao, D. Sun, et al., 2020), CEM (Shao, Yao, et al., 2018), and PEM (Shao, D. Sun, et al., 2020). EM iteratively estimates user reliability and claim truthfulness. Starting with initial assumptions, it calculates claim probabilities and refines estimates through alternating E-steps and M-steps until convergence, optimizing both parameters. EM-MultiF incorporates additional features beyond source reliability, like whether the tweet contains an image or a URL. It aims to leverage these features to improve truth discovery. While PEM penalizes the probability of a claim being false if it has supporting features (like images or URLs). The idea is that claims with more supporting evidence are more likely to be true. Lastly, based on the number of independent features supporting a claim, CEM ensures that the probability of the claim being true is higher when supported by multiple independent sources providing evidence.

These methods exploit information about the associations between "subjects and claims", "features and claims", and the relationship between the claim author and his ancestor in the social graph. These fea-

tures are represented using three matrices: **SC**, **FC**, and **D**, respectively.

Association between Author and Claims: The SC matrix includes the authors of tweets, on the rows, and claims as columns. *SC(i,j)* = 1 if an author *i* posts a tweet related to claim *j*. For example, the SC matrices of two users where both users have posted about claim *i*, and the second user has posted about claim *j* as well, would be represented as:

$$SC_{(i,j)} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

The information needed to execute the algorithm is the informative content of the tweet and the semantic meaning. For example, considering the tweets *Today is Brad Pitt's 57th birthday* and *57 years ago Brad Pitt was born*. Although the two sentences just reported are different from a lexical point of view, the informative content of both is the same. Clustering is used to achieve this aim.

Associations between Features and Claims: The FC matrix contains features about i) presence of images, ii) presence of URLs, and iii) claim reported by at least two independent sources. Such an FC matrix represents the relationship between claims and features, with rows for features (three total) and columns for claims. If claim C_j contains feature F_k, the cell is set to 1; otherwise, it is 0. For example, if the claim in column 6 includes an image, $F_0C_5 = 1$. The same applies to URLs. For the third feature, independent sources reporting similar claims are marked with 1. The matrix uses binary indicators without checking the actual content of images or URLs. The FC matrix with two claims is shown as an example.

$$FC_{(i,j)} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}$$

This indicates that in claim *i*, both images and URLs are present, and multiple independent sources shared similar content. While claim *j* contains URLs only.

Associations between Claim and Author's Ancestors: The D matrix reports the authors on the rows and the claims on the columns. In this case, the information is the relationship between the author's ancestor

and the claim. The cell of the matrix D corresponding to the source S_i and the claim C_j will have the value 1 if there is at least one ancestor of S_i who has reported the claim C_j, as shown below.

$$D_{(i,j)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

3.3 Results and Discussion

The dataset collected using the proposed method consisted of 106,901 tweets and 4301 clusters representing claims. Among these, 57,413 tweets were classified as noise points and not associated with any claim, leaving the remaining tweets for further analysis. Of the 4,301 claims, the Google Fact Check system annotated 4,245 as *Unknown*, 9 as *Undetermined*, 6 as *True*, and 41 as *False*. Therefore, 47 claims containing 667 tweets and 653 users were used for the analysis.

For the comparison of the proposed method, this research employs two publicly available datasets. The first dataset consists of news articles from Politifact and Gossipcop [1], including tweet IDs, URLs, and titles. The second dataset was collected from a Dropbox repository (Shu, Mahudeswaran, et al., 2020). The former exhibits a significant class imbalance, while the latter is balanced as outlined in Table 1. As mentioned above, the fake news algorithms utilize features described in Section 3.2. The dataset generated by the proposed method already consists of these features. X API is used to retrieve detailed data of tweets from the public dataset, which extracts tweet content, metadata, and user information, such as ID, bio, and follower counts. For constructing the SC (source-claim), FC (feature-content), and D (dependency) matrices, the API helps map tweet-author relationships and build an influence graph. Retweets and follower-following data are combined to identify ancestor-descendant relationships between users. The *api.show_friendship()* function is used to verify these relationships, forming the basis for constructing the matrices. The dimensions of feature matrices and their values for

[1] https://github.com/KaiDMML/FakeNewsNet

Table 1: Information about the datasets adopted.

Dataset	Imbalanced	Balanced	Proposed
Users	1554	495	653
Tweets	4285	682	667
True Claims	168	101	6
False Claims	638	135	41
Total Claims	806	236	47
Features	3	3	

Table 2: Feature matrices' dimensions and values.

	Imbalanced Dataset			Balanced Dataset		
Matrix	Dimenison	Number of 1s	Number of 0s	Dimenison	Number of 1s	Number of 0s
SC	1554×806	3246	1249278	495×236	591	116229
FC	3×806	1235	1183	3×236	303	405
D	1554×806	0	1252524	495×236	0	116820

each dataset are shown in Table 2, while Figure 2 shows the count of presence (1) and absence (0) of images and URLs in each dataset.

The four algorithms introduced in Section 3.2 were extensively tested to determine their optimal working parameters. In case of EM and EM-MultiF, *fraction* refers to a parameter that determines the subset of the data to be considered in each iteration of the algorithm. In this case, the fractions for EM and EM-MultiF were analyzed across a range of values from 0.2 to 0.9, with the optimal value found to be 0.9 for both algorithms. For the PEM framework, the parameter α controls the penalty term during the estimation process, thereby influencing the degree of emphasis placed on the source reliability. The optimal value of α found for PEM was 0.8 from values ranging from 0.1 to 0.8. For CEM, the parameter λ is related to the constraints imposed on the estimation process, affecting the weight given to these constraints. The optimal value of λ found for CEM was 0.8.

The models are applied to the two public datasets, as well as to the one generated by the proposed system. The balanced dataset was used

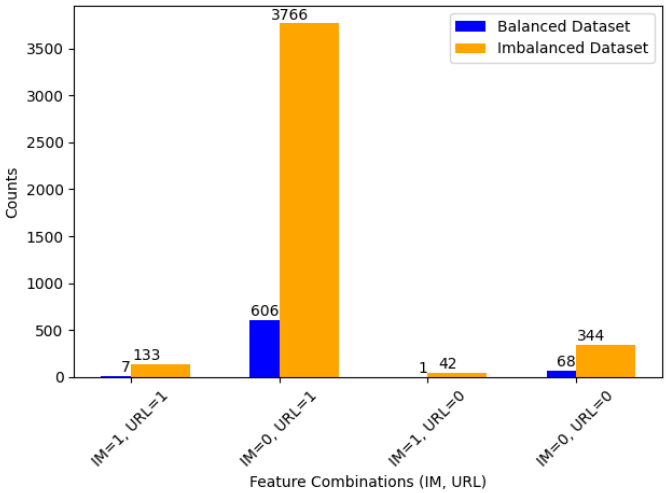


Figure 2: Comparison of Images and URL Combinations in Datasets.

for training the models, and the other two datasets, i.e., imbalanced and proposed, for testing, reflecting the real-world imbalance typically observed in fake news scenarios. As illustrated in Table 3, all models show good performance on the testing datasets. However, the performance is slightly lower on the imbalanced dataset, which could be due to the class imbalance. While the proposed dataset, despite sharing a similar imbalance, includes features and patterns that enhance model performance, it maintains a clear balance between the metrics. A different perspective on the comparison of results obtained across the testing datasets is illustrated in Figure 3. It is also worth highlighting the effectiveness of the proposed data collection method, where all necessary data is readily available, eliminating the need to construct separate feature matrices for evaluating each model and to use Twitter APIs, which can be time-consuming and subject to rate limits. The main advantage of the system-generated dataset over traditional datasets is that it simplifies the process of creating annotated datasets for evaluating and testing fake news detection algorithms.

Table 3: Performance of models on three datasets.

Dataset	Algorithm	Accuracy	Precision	Recall	F1-score
Balanced (train)	EM	0.62288	0.61057	0.94074	0.74052
	EM-MultiF	0.66949	0.64467	0.94074	0.76506
	PEM	0.64406	0.65644	0.79259	0.71812
	CEM	0.65670	0.63910	0.91850	0.75370
Imbalanced (test)	EM	**0.799**	0.83147	**0.93573**	**0.88053**
	EM-MultiF	0.79528	0.83263	0.92789	0.87768
	PEM	0.77295	0.83604	0.88714	0.86083
	CEM	0.74193	**0.84789**	0.82131	0.83439
Proposed (test)	EM	**0.78723**	**0.87804**	**0.87804**	**0.87804**
	EM-MultiF	**0.78723**	**0.87804**	**0.87804**	**0.87804**
	PEM	0.76595	0.87500	0.85365	0.86419
	CEM	0.74468	0.87179	0.82926	0.85000

Additionally, this study offers valuable insights into the performance of fake news detection models based on the obtained results. When comparing EM (user reliability-based) with EM-MultiF (which includes images and URLs), there was little performance improvement from adding content features, suggesting that multimedia presence alone is not a strong indicator of truth. EM and EM-MultiF outperformed PEM and CEM, possibly due to the limitation of the latter in handling complexities in user-claim relationships. It is important to note that the user relationship matrix (D) lacked data, limiting the algorithms' ability to leverage user influence, thus affecting overall performance.

3.4 Summary and Future Work

The primary goal of this research was to create a comprehensive dataset for fake news detection algorithms to help them learn the nuanced patterns of data more effectively. The proposed approach provides a more accurate attribution of truth values, which is a key factor for a reliable model evaluation. The dataset was used to evaluate the performance of four unsupervised models: EM, EM MultiF, CEM, and PEM. For com-

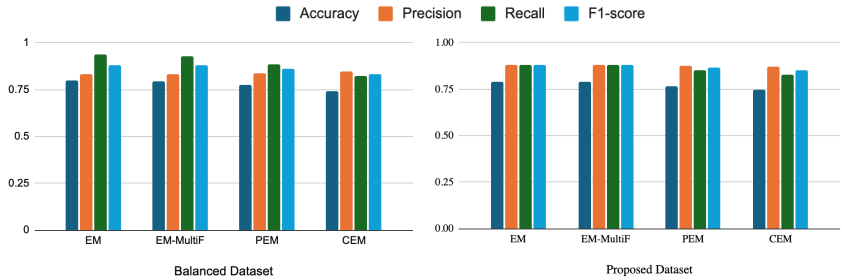


Figure 3: Performance of models on public dataset and dataset obtained by proposed model.

parison, two publicly available datasets were also analyzed. The results demonstrated that the imbalanced dataset and the dataset generated by the proposed method yielded comparable outcomes, with the proposed dataset offering more consistent and balanced metrics. These findings confirm that the proposed method is not only effective for evaluating fake news detection models but also advantageous in producing reliable and balanced results. By streamlining the data collection and preprocessing process, this method enables researchers to utilize a complete and well-structured dataset, saving time and resources while enhancing the accuracy of evaluations. Future work could focus on refining user relationship modeling by incorporating more engagement types (e.g., likes) and exploring alternative data sources. Enhancing data retrieval to include complete user-pair information could improve results.

Chapter 4

Multi-Domain Fake News Detection using Ensemble Learning

Traditional fake news detection models often rely on a single view for classification, overlooking the intricate patterns that fake news contains. A news item may be more representative in certain aspects, such as semantics and emotions, rather than other features like writing styles. Considering only one aspect for all news items can lead to ineffective model performance, reducing the model's ability to generalize and accurately detect fake news. Another limitation in the literature is the insufficient consideration of the domain knowledge to which a news item belongs, neglecting its importance in the classification process. Since fake news often contains domain-specific information, its detection requires contextual understanding, which is frequently overlooked in existing approaches. To address these issues, this work (Batool, Re, Morana, and Tortorici, 2025) introduces an advanced machine learning model-based ensemble technique for accurately recognizing fake news and competing with state-of-the-art models. More precisely, the contributions of this work are as follows:

- To break down each news item into three types of features– seman-

tic, emotional, and stylistic– capturing a comprehensive view of the content.

- To develop an ensemble model that integrates well-established fake news detection models, aiming to maximize classification accuracy.

- To integrate domain knowledge into the classification process, emphasizing the most relevant features in the context of fake news, to obtain more accurate classification results.

4.1 Background and Methodology

4.1.1 Multi-Domain Feature Extraction

Similar to the approach implemented by Zhu et al. (2022), the feature extraction method operates at two levels. First, three types of basic features, i.e., *semantic*, *emotional*, and *stylistic*, are extracted from the text. These are further propagated into three distinct deep extractors to provide higher-level representations. The entire process is summarized in Figure 4, where *semantic*, *emotional*, and *stylistic* features extractions are represented by blue, orange, and green modules, respectively.

Level 0 Features

For **semantic view**, the pre-trained RoBERTa (Robustly Optimized BERT Pretraining Approach) tokenizer is employed, similar to the approach used by Nan et al. (2021). The content of each news item is tokenized using Byte-Pair Encoding (BPE) (Shibata et al., 1999), which preserves common words and breaks down rare or unseen words into sub-units. Each generated token is mapped into a dictionary, with special tokens such as *[CLS]* for the start of the sequence, *[SEP]* for the separation between the sentences, and *[PAD]* for padding to make sure uniform sequence length. A maximum length of 300 is set, reflecting the character length constraint typical on social media platforms such as X. Additionally, an attention mask was also generated to distinguish the actual tokens from

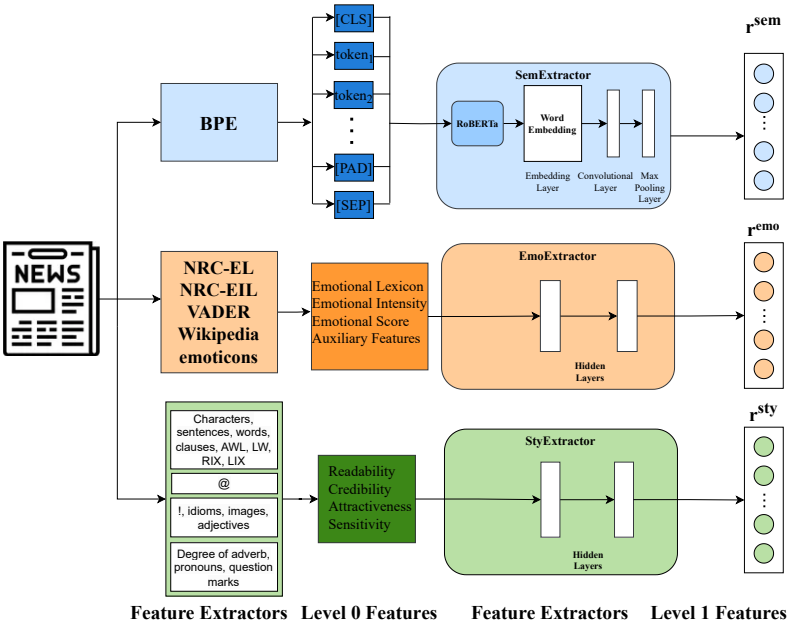


Figure 4: Multi-Domain Feature Extraction

padding, where the mask is set to 1 for real tokens and 0 for padding tokens.

The **emotional view** captures the feeling of both authors' and readers' towards the topic; therefore, integrating this information allows the introduction of useful patterns in the detection of fake news. Here, 38 emotional features are extracted and grouped into four categories:

- Emotional Lexicon: This set captures emotions through text that conveys emotion using specific words. The NRC[1] Emotion Lexicon (NRC-EL) is used, which associates words with eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two feelings (negative and positive). For each word in the lexicon, a flag (with value 0 or 1) denotes its association with a given

[1]https://github.com/RMSnow/WWW2021/tree/master/resources/English/NRC

emotion.

- Emotional Intensity: Each word in this set is given a score (between 0 and 1) depending on the strength of emotion conveyed by it. The NRC Emotion Intensity Lexicon (NRC-EIL) (Mohammad and Turney, 2013) is used in this step.
- Emotional Score: This set measures the impact of emotion related to the text, using numerical values through the Valence Aware Dictionary and sEntiment Reasoner (VADER) (Hutto and Gilbert, 2014) package of the NLTK library.
- Auxiliary Features: The aim is to extract the characteristics of non-verbal elements, such as emoticons, punctuation elements, and capital letters. Emoticons from Wikipedia[2] are utilized in this step.

The **stylistic view** captures the fact that fake news authors have distinct linguistic patterns, i.e., they need to adopt a particular writing style to convince readers of authenticity and achieve high engagement. Therefore, analyzing these patterns can help in the detection of fake news (Vishwakarma et al., 2023). Based on the study by Y. Yang et al. (2019), to study the **stylistic view** of the text, a total of 18 stylistic features were extracted and grouped into four categories of *readability*, *credibility*, *attractiveness*, and *sensitivity* of a text.

Level 1 Features

Level 0 features extracted so far are propagated into deep extractors, called SemExtractor, EmoExtractor, and StyExtractor. The **SemExtractor** module employs a TextCNN model, which receives the Level 0 features as inputs and uses the pre-trained RoBERTa model to generate word embeddings. Convolutional filters are then applied, followed by a max pooling operation to produce deep semantic features. The resulting output is a feature vector r^{sem} having a dimension of 320 (64 feature maps x 5 filters). The **EmoExtractor**, designed for deep emotional representations, utilizes a Multilayer Perceptron (MLP) consisting of three

[2]https://en.wikipedia.org/wiki/List_of_emoticons

layers. The input to the network is the Level 0 emotional features. The first hidden layer expands the initial 38-dimensional vector into a 256-dimensional representation using the ReLU activation function for non-linearity. Then the second hidden layer further expands the feature vector to a 320-dimensional representation. The output is a feature vector denoted as r^{emo}. Similar to EmoExtractor, the **StyExtractor** also consists of an MLP architecture with three layers. The difference is that the initial stylistic feature vector is 18-dimensional. All the resulting feature vectors are 320-dimensional representations of the same news from different views.

4.1.2 Multi-view Ensemble Classifier (MEC)

The proposed model employs a multi-view approach to detect fake news by integrating three types of deep-features (r^{sem}, r^{emo}, r^{sty}). The goal is to analyze each news item from different perspectives to obtain a more accurate classification through an ensemble-based system of classifiers. For each feature group, multiple machine learning models, including Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), and Multilayer Perceptron (MLP), are used to constitute the base learners within the ensemble. It is then followed by soft voting to aggregate the probabilities produced by each ensemble. Lastly, to combine the predictions from the chosen classifiers, three voting strategies have been evaluated, namely *hard*, *soft*, and *weighted voting*.

In the case of **hard voting**, each of the n ensembles provides a discrete prediction $\hat{y}_i$ for each sample x. The final decision is determined by choosing the class $\hat{y}$ that obtains the majority of predictions.

$$\hat{y} = \arg\max_{c} \sum_{i=1}^{n} \mathbb{I}(\hat{y}_i = c). \tag{4.1}$$

In **soft voting**, instead, each of the n ensembles returns a probability $P_i(y = c|x)$ indicating that sample x belongs to class $c \in \{0,1\}$. The final prediction is based on the arithmetic mean of the probabilities provided by each ensemble:

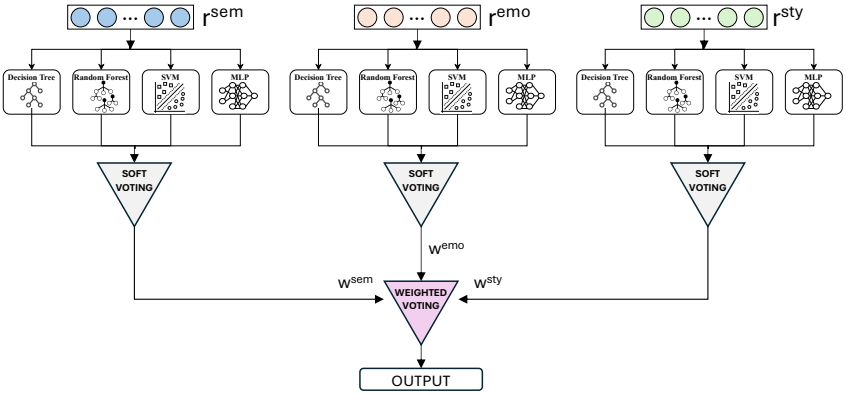


Figure 5: Multi-view Ensemble Classifier (MEC)

$$P(y = c|x) = \frac{1}{n} \sum_{i=1}^{n} P_i(y = c|x). \tag{4.2}$$

Then, the class with the highest probability is selected:

$$\hat{y} = \arg\max_{c} P(y = c|x). \tag{4.3}$$

The **weighted voting** strategy is similar to soft voting, but weights w_i are assigned based on the domain d to which a given sample belongs. The weights are determined during the validation phase, based on the intermediate F1-scores and Accuracy metrics (Wardoyo et al., 2020).

For each domain d, a weight vector $w_d = [w_d^{sem}, w_d^{emo}, w_d^{sty}]$ is created representing the influence of the semantic, emotional, and stylistic views on classification within that domain. Each vector is normalized so that the sum of weights is 1. The vectors obtained for each single domain form a weight matrix W consisting of m rows corresponding to the domains and n columns corresponding to the views. For the three-domain problem, a 3x3 matrix is generated as follows:

$$W = \begin{bmatrix} w_0^{sem} & w_0^{emo} & w_0^{sty} \\ w_1^{sem} & w_1^{emo} & w_1^{sty} \\ w_2^{sem} & w_2^{emo} & w_2^{sty} \end{bmatrix} \tag{4.4}$$

In the proposed experimental scenario, two separate matrices will be considered, one for F1-scores and one for Accuracies. During the final prediction phase, assuming that the news item to be classified belongs to the j-th domain, the corresponding weight vector w_j is extracted from the *W* matrix. The final prediction is then based on the weighted average of the probabilities provided by each ensemble:

$$P(y = c|x) = \frac{1}{\sum_{i=1}^{n} w_i} \sum_{i=1}^{n} w_i P_i(y = c|x) = \sum_{i=1}^{n} w_i P_i(y = c|x). \quad (4.5)$$

The class with the highest weighted probability is selected as the final prediction:

$$\hat{y} = \arg\max_{c} P(y = c|x). \quad (4.6)$$

The weighted voting ensures that the domain characteristics are tailored to each view, leading to an improved classification performance.

4.2 Experimental Analysis and Results

To conduct the experiments, a custom dataset was created by merging data from two existing datasets: FakeNewsNet and MM-COVID.

The *FakeNewsNet*[3] is one of the widely used datasets in research and includes data of 23194 news items from two domains: entertainment *(GossipCop)* and politics *(PolitiFact)*. *MM-COVID* [4] dataset is a multilingual and multimodal dataset designed for news-related COVID-19 pandemic. The news items have been labeled as true or false by reliable fact-checking sources, such as Snopes and the International Fact-Checking Network (IFCN). The dataset consists of news items in six languages: English, Spanish, Portuguese, French, Hindi, and Italian. In particular, there are about 4000 news items in English, equally balanced between real and fake.

[3]https://github.com/KaiDMML/FakeNewsNet
[4]https://github.com/bigheiniu/MM-COVID

Table 4: Parameters for each classifier used in MEC

Classifier	Semantic Features	Emotional Features	Stylistic Features
DT	criterion=entropy, max_depth=10	criterion=entropy, max_depth=10	criterion=entropy, max_depth=10
RF	n_estimators=1000, max_depth=20	n_estimators=1000, max_depth=20	n_estimators=300, max_depth=20
SVM	kernel=poly, C=1, gamma=1	kernel=rbf, C=100, gamma=0.001	kernel=rbf, C=10, gamma=0.001
MLP	activation=relu, solver=sgd, alpha=0.0005, hidden_layer=(150,100,50)	activation=relu, solver=adam, alpha=0.0001, hidden_layer=(100,)	activation=relu, solver=adam, alpha=0.0001, hidden_layer=(100,)

Note that since the datasets are old, most users were unavailable, and due to privacy policies, the complete dataset was not utilized in this research. The comparison with state-of-the-art models is also made according to the available dataset only.

Considering a strong imbalance in the dataset, a sub-sampling was performed to ensure a homogeneous distribution of real and false news. Thus, the final dataset comprised a total of 14000 news items, belonging to three different domains: GossipCop, PolitiFact, and COVID. Each news was characterized by its ID, content, label, and domain. The data distribution was 50% of fake news and 50% of real news. In case of domains, GossipCop was 76%, samples from PolitiFact were 6.2%, and 17.8% of samples from the COVID dataset.

The experiments were conducted with a 60-20-20 train-test-validation split, while a grid search approach was employed to optimize the hyperparameters for each classification model with k-fold cross-validation (k = 5). The optimal hyperparameters identified were then used to train the base classifiers for each feature group. The configurations of the classifiers are as mentioned in Table 4. Additionally, a fixed random seed (random_state=2024) was set to ensure experimental reproducibility.

The first experiment aimed to evaluate the performance of the proposed architecture using a balanced dataset (50% real and 50% fake) of

Table 5: First experiment: balanced dataset and uniformly sampled domains.

	Average				Domain-specific Accuracy		
Voting Strategy	**Acc**	**Prec**	**Rec**	**F1**	**GossipCop**	**PolitiFact**	**COVID**
Hard	0.767	0.759	0.780	0.769	0.572	0.924	0.805
Soft	0.778	0.761	0.811	0.785	0.572	**0.930**	0.833
Weighted (Acc.)	0.788	0.769	0.822	0.795	**0.595**	**0.930**	**0.839**
Weighted (F1)	0.786	0.766	0.822	0.793	0.590	**0.930**	**0.839**

news uniformly taken from the GossipCop, PolitiFact, and COVID domains (i.e., 33.33% from each dataset). The objective was to evaluate the model's performance in the absence of any bias resulting from a non-uniform distribution of samples. The results obtained by applying the various voting strategies are reported in Table 5 with the best metrics highlighted in bold.

As can be observed, the PolitiFact domain was consistently classified with high accuracy (Acc) across all voting strategies, likely because the news belonging to this domain has a more structured nature. Good results are also obtained in the COVID domain, whilst the GossipCop posed a challenge to the model. The reason might be that the news of this domain presents more unpredictability, which would require the model to be trained with a larger number of samples to detect these aspects. From an overall performance analysis, the proposed weighted voting strategies (both based on accuracy and F1-score) emerged as the most balanced and well-performing ones, achieving the highest average (over all domains) accuracy and F1-score values. At the same time, the soft voting strategy also proved to be a good alternative, consistently performing well compared to hard voting, which is the least flexible among the strategies.

In the second experiment, the entire available dataset was used, which showed a balance for the classification labels (50% real and 50% fake) and an imbalance for the domain labels (GossipCop domain (76%), COVID (17.8%), and PolitiFact (6.2%)).

The results, reported in Table 6, are significantly better for GossipCop, with a slight reduction in accuracy for the other two domains. This

Table 6: Second experiment: balanced dataset and unbalanced domains.

	Average				Domain-specific Accuracy		
Voting Strategy	**Acc**	**Prec**	**Rec**	**F1**	**GossipCop**	**PolitiFact**	**COVID**
Hard	0.748	0.766	0.716	0.740	0.745	0.756	0.762
Soft	0.783	0.798	0.757	0.780	0.780	0.791	0.804
Weighted (Acc.)	0.790	0.802	0.771	0.786	0.778	**0.802**	0.836
Weighted (F1)	0.792	0.803	0.772	0.787	**0.779**	0.796	**0.844**

Table 7: Third experiment: unbalanced dataset and unbalanced domains.

	Average				Domain-specific Accuracy		
Voting Strategy	**Acc**	**Prec**	**Rec**	**F1**	**GossipCop**	**PolitiFact**	**COVID**
Hard	0.768	0.756	0.636	0.690	0.649	0.915	0.739
Soft	0.780	0.799	0.616	0.696	0.654	**0.943**	0.744
Weighted (Acc.)	0.796	0.806	0.659	0.720	**0.663**	0.938	0.787
Weighted (F1)	0.812	0.812	0.702	0.753	**0.663**	0.938	**0.834**

turns out to be predictable because of both a significant increase in the number of samples and the unbalanced distribution of the domains. Notably, the higher accuracy observed in the GossipCop domain supports the hypothesis from Experiment 1: a larger volume of samples enables the model to capture more complex patterns, thereby enhancing classification performance

The third experiment presented a more realistic scenario, where real news is present in a greater quantity than false news, i.e., 41% vs 59%. The results are presented in Table 7, which shows that the weighted voting strategies again outperform others across multiple datasets. Here, hard and soft voting strategies provided competitive performance but were unable to capture the nuanced patterns, such as in the GossipCop domain, due to the low number of its samples. While earlier studies (Fumera and Roli, 2005) suggested that the weight estimation might be challenging, the proposed approach addresses this by achieving competitive or better F1-scores and accuracies, demonstrating that the weighted voting performs better than simple voting. The reason is that different domains exhibit distinct properties in terms of semantics, emotional, and stylistic features. This overall analysis suggests that weighted ensemble methods, especially those focused on F1 scores, are well-suited for

multi-domain fake news detection where cross-domain features must be effectively accounted for.

Finally, to demonstrate the effectiveness of the proposed architecture as compared with state-of-the-art, Table 8 compares the overall performance of MEC with the M3FEND model. It is worth noting that M3FEND was tested on the same dataset and features used to evaluate MEC; therefore, the results obtained are slightly different from those reported by Zhu et al. (2022). As can be observed, MEC achieves better performances according to all four metrics, with a few exceptions where values are almost comparable.

Table 8: Comparison with state-of-the-art Zhu et al., 2022

Experiment	Model	Acc	Prec	Rec	F1
Experiment 1	M3FEND	0.769	**0.792**	0.764	**0.804**
	MEC (Weighted Acc. Voting)	**0.788**	0.769	**0.822**	0.795
Experiment 2	M3FEND	0.749	0.754	0.747	0.768
	MEC (Weighted F1 Voting)	**0.792**	**0.803**	**0.772**	**0.787**
Experiment 3	M3FEND	0.743	0.744	**0.748**	0.728
	MEC (Weighted F1 Voting)	**0.812**	**0.812**	0.702	**0.753**

Chapter 5

Blocking Fake News Propagation exploiting OSNs Users Interests and Connections

Online Social Networks (OSNs) are increasingly serving as powerful channels for fake news spreading, often leading to serious consequences, such as economic disruption and public distress (Batool, Canino, et al., 2022). As the underlying mechanisms remain largely unclear, several diffusion models (in which users are represented as nodes of a graph and their interactions as edges) have been developed to better understand the phenomenon. Generally, these models focus primarily on simulating information propagation through social influence, assuming that each node transmits information to its neighbors with a certain probability, allowing the infected ones to disseminate further.

Influence diffusion can be managed as an Influence Maximization (IM) problem, i.e., optimizing the spread of influence through a monotone submodular function. While effective, models that leverage this strategy often oversimplify information diffusion by reducing it to a binary process. In reality, the news spreading within a network depends on

multiple factors, allowing for multiple variations and adaptations. Explanatory models focus on factors such as influential nodes (D. Chen et al., 2012; Gaglio, Lo Re, and Morana, 2015), acceptance criteria (Han and Niu, 2013), and node characteristics (Ou et al., 2017). Predictive models aim to forecast the spread of information and are widely used in influence maximization. Early models, such as Kempe, Kleinberg, and Tardos (2003), faced efficiency problems, leading to improved approaches such as IRIE (Jung, Heo, and W. Chen, 2012) and IPA (J. Kim, S.-K. Kim, and H. Yu, 2013), although these lacked accuracy. Other models (Borgs et al., 2014; Tang, Xiao, and Shi, 2014) were computationally intensive. A key predictive model, the *Independent Cascade* (IC) (Kempe, Kleinberg, and Tardos, 2003), describes how an active node probabilistically influences a neighboring node.

Similar to IC, the *Linear Threshold* model (LT) (Granovetter, 1978)(Watts, 2002) shapes propagation based on cumulative influence thresholds. W. Chen, Yuan, and L. Zhang (2010) developed a faster greedy algorithm, while H. Chen and Y. Wang (2012) introduced a heuristic approach to identify key seed nodes. Both IC and LT, whose use in the context of fake news dissemination is discussed by Shakarian et al. (2015), rely on Monte Carlo simulations, which are computationally expensive. To overcome these inefficiencies, Goyal, Lu, and Lakshmanan (2011) introduced SimPath, which focuses on local neighborhoods to reduce complexity while maintaining accuracy. SimPath optimizes performance using the cost-effective, lazy forward method, which significantly improves scalability over traditional IC and LT models. Accurate determination of edge weights is critical to effective propagation modeling, yet research has largely focused on node classification rather than edge representation. Common techniques include Jaccard similarity (Aggarwal, He, and Peixiang Zhao, 2016) and interaction-based measures (P. Yang and Peilin Zhao, 2015). Hangal et al. (2010) quantified tie strength based on user interactions, showing that the most influential path is not always the shortest. On Twitter, edge weights are often based on retweet counts (Subramani et al., 2011). R. Liu et al. (2014) proposed a method that incorporates user attractiveness and mutual interests to refine edge weight calcula-

tions. Trust and shared interests (Gaglio, Lo Re, and Morana, 2016) play a key role in information sharing and reliability. Community detection methods, such as ISCoDe (Jaho, Karaliopoulos, and Stavrakakis, 2011), identify clusters based on interest similarity and define edge weights accordingly.

This work, as proposed in (Batool, Re, Morana, and Rizzo, 2025), introduces a novel diffusion model that integrates both friendship and mutual preferences to provide a much realistic representation of relationships and a more accurate understanding of the dynamics of information diffusion. Then, a well-established edge-blocking technique was adopted, and its containment ability was compared as to whether the propagation network is modeled with the neighborhood information only or with the proposed model. The idea is to show that considering mutual interests among users helps to better describe the spread of fake news and thus make the blocking algorithm more effective. To this end, extensive experiments were conducted in various scenarios.

5.1 Problem Formulation

A social network can be represented as a directed graph $G = (V, E)$, where V denotes user accounts and E represents their relationships.

A directed edge $(u, v) \in E$ denotes a connection from user u to v, allowing u to access multimedia content shared by v. In modern OSNs, recommendation algorithms play a crucial role by curating content based on user preferences, thus heavily shaping the diffusion process. Since content that does not match a user's interests is less likely to appear in their feed, its distribution is inherently limited. Once exposed to it, users can decide whether to share it or not.

5.1.1 The Network Model

Modeling trust between interacting entities in an OSN is inherently complex due to the decentralized nature of these platforms; its results are particularly important in scenarios involving the dissemination of news,

whether real or fake. The level of trust between two accounts influences the likelihood that one of them will believe the other and share the information within their network. Hence, the problem is how to model the concept of trust to describe the spread within an OSN. While the graph G could represent an OSN, it alone does not convey the level of trust between nodes. Information diffusion strongly depends on the structural properties of G (Buskens, 2020), e.g., users with mutual friends are more likely to form connections, whereas distant ones interact less frequently. More generally, the more neighbors there are between two accounts, the stronger the connection. To quantify this trend, to each edge in G is assigned a neighborhood similarity weight $\omega_f(u, v) \in [0, 1]$, which can be measured using the Jaccard index:

$$\omega_f(u, v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}, \tag{5.1}$$

where $N(\cdot)$ indicates the set of possible outgoing neighbors of a given node. In this scenario, a higher number of mutual friends between two network entities implies a higher level of trust. We define the *Friendship network* (**FNet**) as a weighted graph $G_s = (V, E, \omega_f)$ that integrates the graph G with neighborhood-based data. While social networks reveal the existence of friendships between accounts, the nature of these relationships remains unclear. To address this, a correlation function is introduced to characterize relationship types, based on computed correlation values (Y. Xu et al., 2021). Shared interests between accounts play a central role in the dissemination process (Cheng et al., 2021) and are effectively modeled by that: highly correlated users have similar interests, negatively correlated ones have opposite preferences, while neutral users show no strong relationship. The higher the correlation between two accounts, the more likely an individual is to trust news from a highly correlated source. Conversely, a low correlation leads to skepticism.

To model this behavior, we introduce the *Preference network* (PNet), represented as a fully connected undirected graph $G_p = (V, E_p, \omega_p)$, where each edge in E_p represents a preference connection between two accounts, while ω_p defines their preference-similarity, whose calculation is challenging. Given that OSNs provide news across different topics (e.g.,

music, sports), we can leverage a preference matrix (PM) to capture the topic-specific interests of each user. A widely used approach is the pairwise comparison method, which evaluates alternatives based on specific criteria (Cook, 2006). In our approach, a user's preferences, over n alternatives, are encoded in an object-to-object matrix $\text{PM}_u = (pm^u_{ij})_{n\times n}$, where:

$$pm^u_{ij} = \begin{cases} 1 & \text{if topic } i \text{ is above subject } j \\ -1 & \text{if topic } i \text{ is below topic } j \\ 0 & \text{if } i = j \end{cases} \tag{5.2}$$

The acquisition of PMs can be accomplished by using questionnaire surveys (B. Liu et al., 2019), pre-determined approaches (Gong et al., 2020), or analyzing the interactions on the social network itself (Milovanović et al., 2019). Once the PMs are given for all the accounts, the rank correlation can be calculated to describe the degree to which individuals agree with their preferences (W. Guo et al., 2021). Let PM_u and PM_v be the preference matrices of the pair (u, v), then their correlation is:

$$\tau(PM_u, PM_v) = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} pm^u_{ij}\, pm^v_{ij}, \tag{5.3}$$

where the denominator works as a normalization factor to ensure that any given ranking is fully correlated with itself, giving $\tau(PM_u, PM_u) = 1$ (Emond and Mason, 2002). Since the correlation value $\in [-1, 1]$, then for each pair (u, v) with $u \neq v$, it is possible to state that $\tau(PM_u, PM_v) > 0$ means that (u, v) are positively correlated, and they are negatively correlated if $\tau(PM_u, PM_v) < 0$, otherwise they are uncorrelated.

The *Friendship-Preference Network* (FPNet, Figure 6) is ultimately created by merging the FNet and PNet graphs. The set of nodes is preserved, representing the original social media accounts (V), while the set of edges is restricted to that of G_f, since it results from the intersection of $E \cap E_p$. This reduces the computational overhead, as weights are only computed for **physically** connected nodes, while unconnected ones are ignored. Regarding the weight of each edge, the final trust ω_{fp} between $(u, v) \in V$ is composed of ω_f and ω_p according to:

$$\omega_{fp}(u, v) = \alpha\omega_f(u, v) + (1 - \alpha)\omega_p(u, v), \tag{5.4}$$

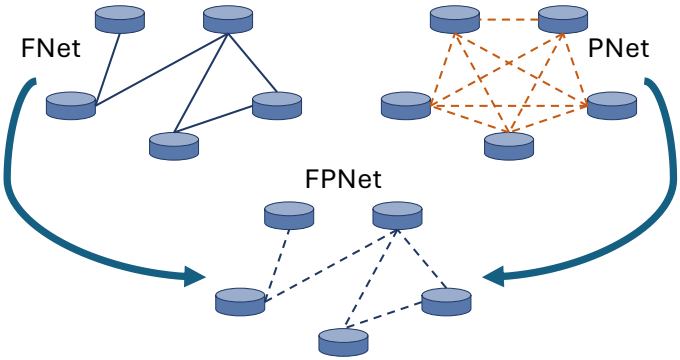


Figure 6: Representation of the Friendship-Net (left), Preferences-Net (right), and the combined FPNet resulting from the proposed approach (center).

where α is the adjustment parameter of the trust degree, satisfying $\alpha \in [0, 1]$. The higher the parameter, the more weight is dependent on the neighborhood; conversely, a greater importance is given to the preference. The construction of the FNet has a complexity of $\mathcal{O}(|E_f|)$, assuming efficient data structures, as it requires computing the Jaccard similarity only between nodes connected by edges in G_f. The preference matrix requires $\mathcal{O}(n^2)$ operations, where n is the number of topic alternatives, although, due to symmetry, this can be reduced to $\mathcal{O}(n^2/2)$. The computation of the pairwise correlations between user preference matrices adds a cost of $\mathcal{O}(|E_p| \cdot n^2)$, resulting in a total PNet construction cost of $\mathcal{O}(|V| \cdot n^2/2) + \mathcal{O}(|E_p| \cdot n^2)$. Finally, the FPNet can be constructed with an additional cost of $\mathcal{O}(|E_p|)$.

5.1.2 Propagation Algorithms

News propagation, in an OSN, is represented by the state of each node - either infected (having received and trusted a rumor) or not. As mentioned above, this issue is analyzed by modeling diffusion influenced by social dynamics, where highly impactful users amplify the spread. Among the well-established models, we focused on the Independent

Cascade (IC) (Kempe, Kleinberg, and Tardos, 2003) and the Linear Threshold (LT) (Granovetter, 1978; Watts, 2002). The first one assumes that each node has n independent chances to become infected, with n being the number of its infected neighbors. At t=0, a random set of nodes, called *seeders*, are initially activated, while at a generic stage t each seeder independently tries to activate its out-neighbours with a certain probability. Once activated, a node has only one opportunity to influence its neighbors before remaining permanently active, and the process continues until no further activations are possible.

In contrast, the LT model determines information "adoption" based on the influence of already infected neighbors. Each edge (u, v) has a non-negative weight representing it, while each node has a threshold θ_u that determines whether it becomes infected or not. The infected nodes attempt to activate their neighbors based on the following activation function:

$$\sum_{v \in N(u)} \omega_{fp}(u, v) \geq \theta_u^{LT}. \tag{5.5}$$

The Linear Threshold model provides a more realistic representation of the entire process than the Independent Cascade, by considering this kind of influence: it captures complex social dynamics and can inform more targeted *law enforcement strategies*. However, LT is computationally more demanding, especially for large datasets. Recall that both IC and LT typically require Monte Carlo simulations to estimate a node's influence over a sufficient period of time. While effective, this approach is time-consuming and impractical for large-scale social networks. To address these limitations, researchers have proposed various optimizations, such as SimPath (Goyal, Lu, and Lakshmanan, 2011), which enhances efficiency through several optimizations, including the *CELF Method* and *Look-Ahead Optimization*. The former minimizes influence-estimation calls by incorporating *Vertex Cover Optimization*, reducing the number of nodes to evaluate. The latter speeds up propagation estimation by predicting the influence of potential seed nodes, making the selection process more efficient as the number of seed nodes increases.

5.1.3 Blocking Algorithms

Understanding how information spreads is essential to understanding network dynamics, but equally important is analyzing the network structure to reduce the spread of harmful content. Therefore, many studies focused on how to contain information spreads by modifying the network (Kuhlman et al., 2013; Budak, Agrawal, and El Abbadi, 2011). **Source-aware approaches** (Shelke and Attar, 2019) identify the source of the infection and aim to reduce its impact on the network, but this is either time-consuming or challenging. On the other hand, the **source-ignorant approaches** (Zareie and Sakellariou, 2022) disregard the source of infection and focus on minimizing the flow of information through the network. Among the source-ignorant approaches, we concentrated on Iterative Efficient Edge Detection (IEED), which aims to block the most critical edges by assigning weights to each edge based on the interactions between nodes. According to Zareie and Sakellariou (2022), IEED computes the blocking efficiency of each edge, which ensures that removing one prevents the spread of infection between connected nodes through their common neighbors.

For each node, its influence is calculated by considering the weights of its neighboring nodes using an entropy-based approach outlined by Zareie, Sheikhahmadi, and Fatemi (2017). The criticality of each edge is then determined by combining the influence with the blocking efficiency of the edge, which effectively mitigates propagation by targeting the core points in the network. These strategies help to effectively contain the spread of harmful information, providing a more immediate response to potential threats on the network.

5.2 Experimental Analysis

The experiments were conducted using the Twitter network dataset (Fink et al., 2023), well-known in this research niche, that captures Twitter interactions among members of the 117^{th} United States Congress between February 9, 2022, and June 9, 2022. The dataset consists of a graph with

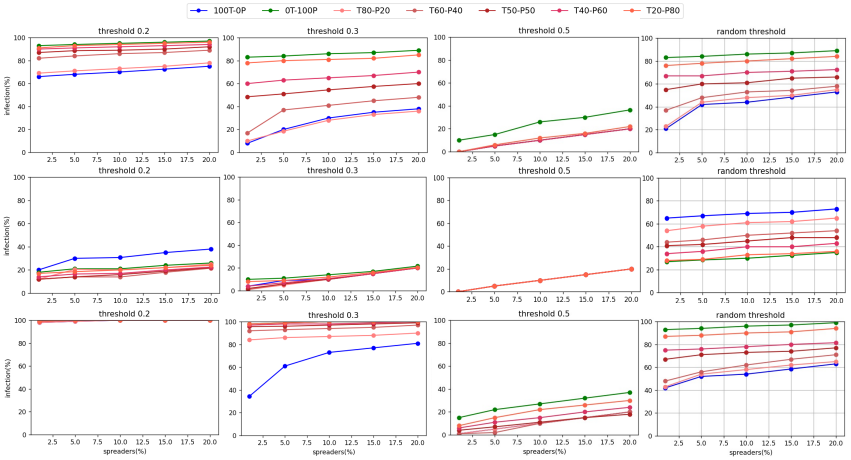


Figure 7: Trend of infection rate as the threshold and spreaders change for IC (top row), LC (middle), and SIMPATH (bottom) models. In each plot, different curves represent different weight distributions between friendship (T) and preferences (P).

475 nodes and approximately 13,000 edges. Although it includes weights representing "probabilities of influence", we also considered additional weights reflecting the users' preferences. For the experiments, preferences were simulated using the approach in (Y. Xu et al., 2021) and tests were performed on the synthetic dataset obtained by adding the generated *preference* layer to the real connections. However, both types of information can be available in any way from the OSN owner, who knows the connections between its users and can calculate preferences with any of the available approaches. The simulated approach is therefore valid and functional for supporting the experimental evaluation.

5.2.1 Analysis on the Diffusion models

The behavior of the diffusion models has been evaluated under three different weighting schemes: (i) friendship-based, (ii) preference-based, and (iii) our combined approach considering different weights, namely 80% friendship-20% preferences (denoted as T80-P20), 60% friendship-

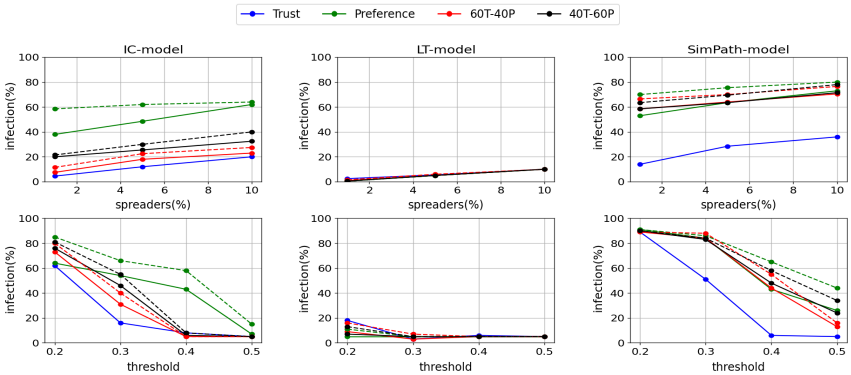


Figure 8: Performance after applying the edge blocking method. The dashed lines show the spread of fake news after the removal of the arcs, considering only the trust, while the solid lines show the spread after the removal of the arcs, considering also the preferences.

40% preferences (T60-P40), their symmetric duals, and a 50-50 approach. In addition, the propagation algorithms will be evaluated based on two key parameters: the number of seeders and the activation threshold θ. The former determines the initial set of active nodes before diffusion begins, while the latter defines when a node becomes infected, depending on the chosen propagation model. It also reflects a user's tendency to believe in news: lower values indicate high trust among users, while higher values indicate skepticism. For all the experiments, we consider $\theta_u^i, i \in \{IC, LT, SP\}$, to be within the range $[0.1, 0.5]$. Values below this range are unrealistic, as they imply no shared friendships or interests, while values above 0.5 suggest extreme skepticism, making message spreading almost impossible.

The first model under analysis is the IC, for which results are shown in the first row of Figure 7. At a low threshold $\theta^{IC} = 0.2$, the propagation algorithm is highly efficient, even with a small number of spreaders, as users readily believe the news. However, as the threshold increases, the success of the diffusion declines. At $\theta^{IC} = 0.3$, the effects of different weighting schemes become more apparent. In general, a higher preference weighting leads to a higher diffusion, as users are more likely to

engage with news that matches their interests.

When the preference-based weighting reaches 100%, the spread is maximized, as news propagates purely based on shared interests rather than direct connections. At higher thresholds ($\theta^{IC} = 0.5$), this trend is even more pronounced, with overall spread decreasing. To simulate a more realistic scenario, in a final experiment, we randomly assigned thresholds for each pair in G_{fp}, given that real-world users have different levels of skepticism.

As expected, the results confirm that a higher number of spreaders leads to an increased infection rate. In the LT model (middle row of Figure 7), the diffusion pattern contrasts with that of IC. Since a node's activation depends on all its active neighbors, diffusion is less effective when weights are heavily preference-dependent. This is because shared interests between neighbors are less common, making activation unlikely. In contrast, when friendship-based weighting is emphasized, nodes with many mutual connections are more likely to activate each other, leading to greater propagation. The aggregating nature of this method is what causes the infected rate to increase dramatically at lower thresholds (Eq. 5.5). When considering $\theta^{LT} = 0.2$, the infection rates range from 10% (when preference is prioritized) to 45% (when only friendships are considered). The number of initial spreaders has little effect, causing only a feeble increase in infection rates. As the threshold increases, spread declines, reaching near zero at $\theta^{LT} = 0.5$, as the activation condition in Eq. 5.5 becomes harder to satisfy. When thresholds are randomly assigned, the diffusion trends are similar to $\theta^{LT} < 0.2$, further confirming that stronger friendship ties enhance diffusion.

The results for SimPath propagation (shown in the bottom row of Figure 7), conversely, reveal several key insights:

- At $\theta^{SP} = 0.2$, the infection rate reaches saturation for all weighting schemes, due to the inherent nature of the propagation mechanism.
- As the threshold increases, disparities between weighting systems become more apparent, first emerging at $\theta^{SP} = 0.3$
- Among the three models, SimPath reaches a surprisingly high dif-

fusion rate at $\theta^{SP} = 0.5$, further emphasizing its effectiveness.

- Moreover, randomly assigned thresholds exhibit behavior akin to fixed ones within the range $[0.3, 0.5]$, while once again, increasing the number of spreaders leads to a gradual increase in the number of infected across all the evaluated scenarios.

An overall evaluation of the results discussed so far can be made by focusing on the cases where only friendship (100T-0P), only preferences (0T-100P), or both factors with 50-50 weights are considered. In all cases, infection capacity improves as the number of spreaders increases, although the magnitude of the improvement varies from model to model. In the LT model, increasing the threshold drastically reduces the spread, while in IC and SimPath, the reduction is more gradual. In particular, the latter is the only one where the spread starts to decrease significantly from $\theta = 0.3$. Another key observation is that in LT, curves converge at $\theta = 0.5$, while in IC, this is true except for the preference-based one. In SimPath, they reach different minimum values, which depend on its underlying selection mechanism, prioritizing paths with the highest marginal gain in influence spread. Finally, in both SimPath and IC, the preference-based weight consistently achieves the highest diffusion rates. This highlights the importance of considering preferences alongside friendship and user interactions when investigating the spread of fake news in realistic scenarios. It reinforces our confidence in integrating both factors for a more comprehensive network analysis.

5.2.2 Impact of the Blocking Algorithms

The next set of experiments aimed to evaluate the performance of a containment algorithm, based on the selective removal of arcs, when dealing with different information propagation methods. Specifically, 20% of the arcs deemed critical by the IEED algorithm are removed for each propagation model. The selection of critical arcs varied according to the different weighting applied, as the balancing factor α influenced the assignment of weights to the edge-blocking algorithm, resulting in different removal configurations. Observing the first column of Figure 8, it can be

seen that, after applying the edge-blocking algorithm, the propagation of IC is reduced as compared to Fig. 7. In particular, configurations in which the preference weight is higher lead to more effective containment. This is because the blocking algorithm, also considering the latent aspect of preference, removes critical arcs in a more targeted way. Furthermore, it can be seen that when the threshold is low, e.g., 0.2, the dissemination capacity is always high because, even if 20% of the arcs are removed, the network is very dense and the news is still able to propagate. Conversely, with the maximum threshold, there is little diffusion; thus, by cutting the most critical arcs, containment has the maximal effect. The trend of LT (second column of Figure 8) is opposite to that of IC, and configurations with higher friendship weights lead to higher diffusion. Since initially the diffusion was already low or zero (see second row of Figure 7), after removing the critical arcs, it is completely canceled, and all options behave the same way. Finally, the containment in the case of SimPath (third column of Figure 8 is similar to that of IC. The main difference is that since SimPath allows a more extensive dissemination than the other methods, the overall containment of fake news is less effective.

A relevant aspect that emerged from the analysis concerns the influence of different weighting strategies. Those who rely solely on friendship were less effective in reducing the spread of fake news than those who also take user preferences into account. In fact, incorporating the latter factor allows for a more precise identification of the nodes in the network from which false information tends to spread more quickly, allowing for a more targeted application of containment. Furthermore, the additional computational overhead is limited exclusively to the construction of the FPNet, as neither the propagation nor the blocking algorithms are impacted by the integration of supplementary information in the weighting scheme. Adopting a model that combines both aspects emerges as the most effective strategy for limiting the spread of fake news, as it allows for more accurate identification of critical nodes and maximizes the effectiveness of the edge-blocking algorithm.

5.3 Summary and Future Work

In this work, we proposed a novel approach to limit the spread of fake news in online social networks (OSNs) by exploiting information readily available to platform owners. A key finding of our work is that OSN owners can significantly improve their ability to mitigate misinformation by adopting a hybrid approach that incorporates both preference-based and friendship-based weighting within the network graph. This method allows for a more targeted and effective mitigation strategy, outperforming traditional approaches that rely solely on friendship or direct user interactions. The experimental analysis underscores the effectiveness of this integrated weighting system, demonstrating that the usage of both systems enables a more adaptive and precise intervention against the spread of fake news. By leveraging both preference-driven behavior and friendship dynamics, our methods provide a scalable and actionable solution to one of the most pressing challenges in online information dissemination. Future research will focus on refining our approach by incorporating machine learning techniques to dynamically adjust weighting parameters in real time, alongside the integration of privacy-preserving mechanisms to protect user preference data.

Chapter 6

Efficient Adversarial Attack on Fake News Detection Systems

To mitigate the manual labor of fact-checking and misinformation detection, both the scientific and industrial communities exploited Artificial Intelligence (AI) and Machine Learning (ML) techniques, enabling the timely detection of potential fake content and triggering fact-checkers only for the most uncertain material. However, most ML-based fake news detection techniques (Shu, Sliva, et al., 2017) share a common structure, utilizing Natural Language Processing (NLP) techniques such as data pre-processing and word embedding, along with Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The idea is that a malicious entity tends to adopt a particular style while writing, with the intent of convincing as many people as possible about the veracity of a statement. In this scenario, features representing the text under analysis, its title, or even the author are analyzed to distinguish genuine and fake content. Lexical features, overemphasized words, or the absence of the information source may also indicate the falsity of the news (Rashkin et al., 2017). However, recent studies (Biggio and Roli, 2018) have proven that it is possible to deceive AI models by adding a

certain amount of perturbation to the input, which causes the classifier to make an error in the final prediction. This concept falls under the domain of Adversarial Machine Learning (AML), which encompasses strategies specifically designed to compromise the reliable functioning of a machine learning algorithm. Regardless of the application scenario they address, AML attacks can be classified as operating in a white-box or black-box setting. The former guarantees a higher probability of success because it assumes that the attacker knows everything about the model to be targeted; in contrast, the latter typology is much more complex and requires the attacker to query a given model with some input to estimate its behavior. In the broader landscape of AML, researchers are making substantial efforts to understand and fortify machine learning models against deliberate attempts. However, amidst this progress, a noteworthy observation is that the literature on adversarial attacks to ML-based fake news detection is quite limited (Ali et al., 2021). The proposed study focuses on this domain, examining the impact of machine learning adversarial attacks on fake news detection models. In particular, this paper presents a black-box attack against an unknown NLP algorithm adopted by the target online platform to identify fake content. A naive approach would require producing several malicious samples and testing whether the system classifies them as genuine or malicious (Ali et al., 2021). However, such a strategy presents two main limitations: first, excessive querying could be interpreted as a potential attempt to compromise the entire system, leading to the blocking of the attacker's account; second, the process is highly time-consuming, as the attack requires a significant number of iterations before success. To overcome these limitations, the proposed approach (Batool, Canino, et al., 2022) leverages a local model to evaluate the validity of malicious samples offline, thereby eliminating the need for repeated interactions with the target system.

6.1 Background and Methodology

The term *Adversarial Machine Learning* is generally referred to as a set of techniques that aim to compromise the proper functioning of ML systems through the use of malicious inputs called *adversarial examples*. In the case of a classification task, the goal is to fool the targeted ML model by obtaining an output different from the expected one.

The guidelines of every AML strategy require the definition of the adversary model according to three main aspects (Gaglio, Giammanco, et al., 2021; Concone et al., 2024).

Firstly, the **attacker's goal** defines the expected result of the attack and in which phase it must be launched. Multiple objectives may be targeted by the adversary, such as *confidentiality* (if the attacker aims to obtain private information), *integrity* (if the aim is to cause the malfunction of the target model), and *availability* (if it is to make the system offline). Since a piece of news is public and the attacker wants to deceive the platform to share and spread fake news, in the scenario considered here, an *integrity violation* is the only objective, e.g., fooling the classifier at the platform's disposal without disrupting the platform itself. In doing so, the proposed strategy can be categorized as a *label-targeted attack* because the attacker wants to maximize the probability that adversarial examples get classified in a specific class, i.e., *real* news. This can be achieved during the *inference* phase, where the crafted sample is evaluated.

Then, the **attacker's capability** defines which values of the original sample are to be perturbed by the attacker, and how. The process of perturbing values is a crucial aspect of the attack: the more samples are altered, the less they will be similar to the original ones, making our adversarial examples useless for the attacker's goal. In the scenario considered here, the attack logic translates into perturbing the original sequence of words X (related to a *fake* news) with some noise, δ, so that the new sequence $\tilde{X} = X + \delta$ is misclassified as a *real* news. To achieve this aim, a set of S_x important words ordered by their influence is identified. One by one, the attacker replaces these words with their disrupted version while maintaining semantic and grammatical similarity to the initial text.

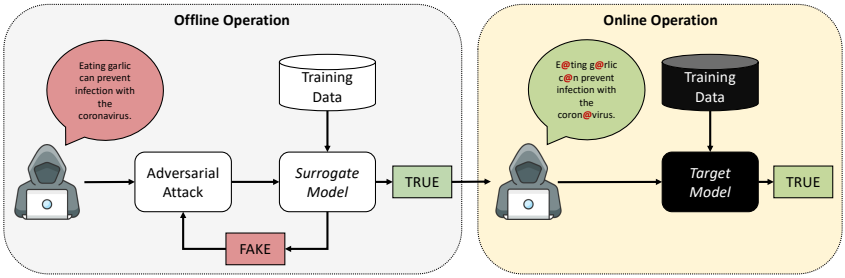


Figure 9: Proposed AML strategy.

Last but not least, the **attacker's knowledge** defines the background the attacker has about the target ML model. The attack is based on the *white-box threat model* when complete knowledge about the target is provided, including the dataset used during the training phase and the model's structure. In contrast, the *gray box threat model* is employed when the attacker has partial knowledge of the target. In real applications, the *black-box threat model* is often adopted. This is the case of the scenario addressed here, as the attacker has no information about the data, the inner ML model, or any other element used by the target. The only way to interact with the remote model is to use it as an oracle for text classification.

However, continuously querying the remote platform could trigger security mechanisms to the point of compromising the attacker. For example, looking at a social network platform, continuously posting news to find out its assigned class could be detected as spamming activity, almost certainly leading to account banning or blacklisting. Similar discussions can be made in other contexts, such as the fact-checking tool provided by Google. In this case, even more drastic measures could be taken since the activity could be interpreted as a DoS attack.

The limits introduced above are overcome by the proposed strategy, following the general structure of black-box adversarial attacks. As shown in Figure 9, the attack is divided into *Offline* and *Online* operations.

In the first phase, the attacker trains a surrogate model S so that it emulates the behavior of the target model T. Note that the application sce-

nario does not impose any special requirements for the surrogate model; however, according to reports in the scientific literature (Biggio and Roli, 2018), the adoption of low-complex ML models (such as a linear regression or support vector machines) allows one to increase the probability that the attack is successful on the target, especially when it consists of convolutional or recurrent neural networks.

To collect a valid dataset for S, the attacker may take advantage of different strategies discussed in the field of AML (Papernot et al., 2017). In addition to these, the attacker could also independently assemble a dataset by exploiting well-known *Fact Checking* sites. It is important to note that the adversary does not need to know the feature space representation for the input samples. Indeed, perturbations can be applied directly to the input text, which in turn will be submitted to the system for classification. Once the model training is completed, the adversary begins to perform the actual *offline operation*, which consists of locally querying S until an erroneous label is returned.

In the second phase, the perturbed text is passed as input to T, and the *transferability* is evaluated. According to this property, it is possible to attack a machine learning system with no knowledge about the underlying model. If the attack is transferred successfully, the target's output will be the same as the surrogate.

6.2 Adversarial Attacks for Text Perturbation

The adversary's objective is to introduce perturbations into the corpus by altering numerous words. The goal is to impact the predictive capabilities of the model without significantly changing the semantic meaning of the text. To accomplish this, the adversary identifies pivotal words within the corpus and implements modifications, such as character replacements or substituting the entire word with its synonym. In particular, the following five attacks were considered:

- **The Probability Weighted Word Salience (PWWS)** (Ren et al., 2019): This attack preserves the lexical, grammatical, and semantic con-

straints of the input while performing the attack. The approach operates by evaluating the importance of individual words, called "word salience," to establish their ranking based on this metric. Subsequently, the approach identifies the word with the highest salience score and compiles a list of prospective substitutions for this word, typically encompassing synonyms or lexically and semantically similar terms. This substitution process is done to alter the model's predictive outcome.

- **Text-Bugger** (Li et al., 2018): Identifies the most important sentences in the text, and for each, an importance value is assigned through a score function. After, variations of the sentences are computed to obtain a new score value to be compared to the original one. The differences between the original score and those obtained from the text's variations are used to determine a set of keywords. This set represents the starting point of the method because it is used to generate five perturbations: random removal of a character, swapping of unique characters, substitution of a character with a homoglyph, random insertion of a whitespace, and substitution of the word by a semantically similar word. Therefore, the attacker chooses the optimal *perturbation* for each keyword in order to reduce the classifier's output score.

- **Text Fooler (TF)** (D. Jin et al., 2020): The approach identifies keywords by computing the difference between the model's score before and after the deletion of a word from the input. The attacker then replaces every keyword with words that are closer to the actual word in a predefined Embedding space and selects the best, i.e., the one that reduces the output score the most.

- **Input Reduction (IR)** (Feng et al., 2018): This attack, in contrast to others, completely deletes the less important words. The result of doing this causes the remaining words to appear nonsensical to humans, and the model also interprets these words as unimportant. Iterative removal of words affects the performance of the model.

Table 9: Admissible perturbations for each of the considered attacks. TextBugger(TB), TextFooler(TF), PWWS, InputReduction(IR), DeepWordBug(DWB)

	TB	TF	PWWS	IR	DWB
Character removal	✓	X	X	X	✓
Character substitution	✓	X	X	X	✓
Character swap	✓	X	X	X	✓
Whitespace insert	✓	X	X	X	X
Semantic similarity	✓	✓	✓	X	X
Syntactic similarity	X	X	X	X	X
Word deletion	X	X	X	✓	X

- **Deep Word Bug (DWB)** (J. Gao et al., 2018): This attack identifies the critical tokens by using a unique scoring strategy. And then perform character-level perturbations, including character swap, character insertion, character deletion, and character substitution, making the words unidentified. These perturbations in the input text impact the model's performance.

For the sake of clarity, Table 9 summarizes the perturbations that each of the considered attacks can make on the original text.

The assessment criteria for the success, failure, or skipping of an attack are contingent on its ability to modify the output label. An attack is called *Successful* when it makes a modification that results in a change of the output label. Conversely, if the attack makes textual perturbations that are not capable of changing the output label, it is categorized as *Failed*.

6.3 Experiment Analysis and Results

Experiments have been carried out on *LIAR* [1], a data set consisting of 2.8K manually labeled short statements. The samples are collected from *Politifact.com*. Each news article encompass various attributes including the *ID*, *label*, *statement*, *subject(s)*, *speaker*, *speaker's job title*, *state info*, *party*

[1]https://www.kaggle.com/datasets/csmalarkodi/liar-fake-news-dataset

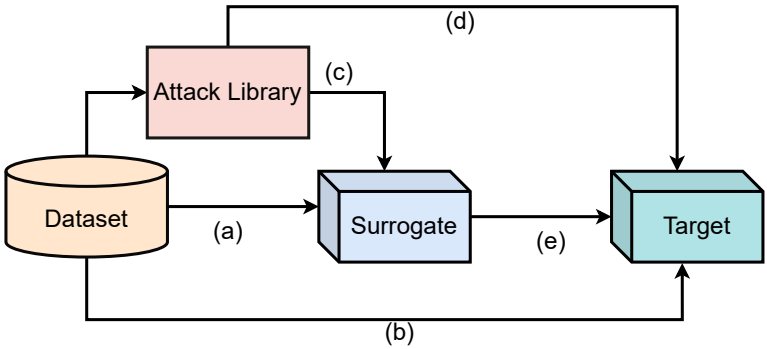


Figure 10: The experimental workflow to assess the performance of the proposed methodology. (a) Evaluation of the surrogate model without launching any attack. (b) Evaluation of the target model without launching any attack. (c) Analysis of the surrogate model on data generated by the attacks. (d) Analysis of the target model on data generated by the attacks. (e) Evaluation of the target model on data perturbed by the surrogate model

affiliation, *barely true counts*, *false counts*, *half true counts*, *mostly true count*, *pants on fire counts* and *venue*. This dataset has been extensively utilized for both binary classification (Bhutani et al., 2019) and multi-label classification (Rasool et al., 2019) tasks in the domain of fake news detection

The dataset underwent preprocessing to convert the raw text into a numerical format computable by machine learning models. The *statement* column served as the input variable and the output variable underwent a transformation where multi-label categories were converted into binary labels denoting either 0 (*Fake*) or 1 (*Real*). This conversion was implemented by associating the labels {*barely-true*, *false*, *pants-on-fire*} to the class 0, and the labels {*true*, *mostly-true*, *half-true*} to the class 1. The *statement* column was also processed to improve the recognition performance. First, the special characters are removed from the news samples, retaining only alphanumeric values. Then, the words are reduced to their base form using WordNetLemmatizer. As stemming does rough reduction and often reduces words to non-words, lemmatization works better by producing valid words and a more accurate representation.

Table 10: Classification performance of the surrogates on D^S and of the target on D^T.

	Surrogate			**Target**
	LR	SVM	RF	NN
Accuracy	.62	.62	.61	.57
Precision	.61	.61	.60	.56
Recall	.60	.60	.59	.55
F1-Score	.60	.60	.59	.55

To simulate a real black-box scenario, two different representations were used for the surrogate and target models. Specifically, we have adopted *Term Frequency-Inverse Document Frequency* (TF-IDF) for the surrogate and the *CountVectorizer* technique for the target model.

In the following section of experiments, Logistic Regression (LR), Random Forest (RF), and Support Vector Machine (SVM) were utilized as surrogate models and a Neural Network (NN) as the target model. In the last case, the input data is passed through an embedding layer to convert words into relevant embeddings. Subsequently, the output is flattened and passed through a dense layer comprising 128 units, the output of which is passed to another dense layer containing 64 units. Finally, the data is sent through a dense layer with two units, as the classification is binary. The ReLU activation function is used for the first two dense layers, while the last layer uses the Sigmoid function.

The efficiency of the proposed methodology has been evaluated following the flow depicted in Figure 10.

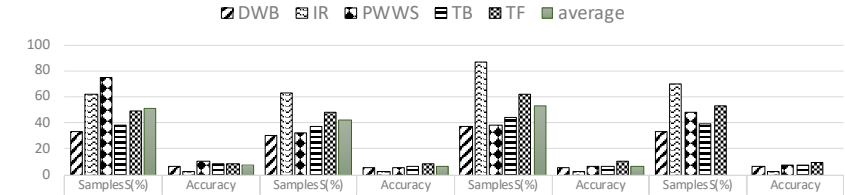
The first set of experiments aims to evaluate the performance of the three surrogate models (configuration (a)) and the target model (configuration (b)) without launching any type of attack. This evaluation allows to determine how the several models perform in the recognition process. The outcomes reported in Table 10 suggest that LR, SVM, and RF achieved similar performance in terms of the considered evaluation metrics, as well as the NN. These values reveal that the different embedding techniques adopted into the ML models do not affect the classification performance. The low recognition efficacy on the LIAR dataset is known

Figure 11: Performance variations of models (target and surrogates) after each attack.

in the literature (Choudhury and Acharjee, 2023), as this dataset contains mislabelled data.

More interesting findings can be deduced for the **configurations (c)** and **(d)**, where the attacks are launched against the models. Here, the *Accuracy Difference*, achieved by each model before and after the attack, is measured: the greater the difference, the more successful the attack. The outcomes presented in Figure 11 indicate that the introduction of adversarial samples leads to a decline in the performance of each model. The most effective attacks are PWWS, TF, and TB, which, on average, show a significant difference in models' performance. All three attacks share the same logic of perturbation strategy, as the semantic similarities of the words are exploited. This observation implies that these attacks are more likely to impact the model's performance. In contrast, strategies that primarily focus on character-level perturbations might have perturbed words such that they go unrecognized by the tokenizer, and possibly filtered out without causing a substantial impact on the model's performance. Other interesting insights concern the number of queries made by each attack procedure. In fact, to launch a successful attack, each time the technique crafts the original text, the target model is queried. This aspect will be examined later in the proposed work.

Finally, the **configuration (e)** considers the entire elaboration pipeline: the samples that fooled each surrogate by each attack are transferred to



	LR		SVM		RF		average	
	Samples%)	Accuracy	Samples%)	Accuracy	Samples%)	Accuracy	Samples%)	Accuracy
DWB	33	7	30	6	37	6	33,3	6,3
IR	62	3	63	3	87	3	70,7	3,0
PWWS	75	10	32	6	38	7	48,3	7,7
TB	38	9	37	7	44	7	39,7	7,7
TF	49	9	48	9	62	10	53,0	9,3
average	51,4	7,6	42,0	6,2	53,6	6,6		

Figure 12: Percentage of samples that fooled the target and Accuracy Difference for each setting in the **configuration (e)**.

the target to evaluate the model's resilience to adversarial examples. In this experimental evaluation, the number of samples that fool the surrogate and the *Accuracy Difference* are examined. The first metric provides objective data on how many samples fool the surrogate. The second metric, on the other hand, provides information on the effectiveness of the proposed technique by demonstrating that the adoption of the surrogate still allows adversarial attacks to be conducted to deceive the target. In this regard, to conduct an effective evaluation, the difference in accuracy was calculated by deriving the deviation between the target's accuracy on the original test set and that achieved by the same model when the test set contains the perturbed samples.

Figure 12 shows the number of samples that fooled the surrogate models and the accuracy achieved by the target for each of the considered settings. As already observed in Figure 11, the introduction of perturbed samples leads to a subsequent decline in the accuracy of the target model. In general, it is important to note that all the surrogates on average (last row) achieve similar performance when transferred to the target model (from 6.2 to 7.6). Here, LR is the best as it shows a trade-off between the metrics considered. Considering the specific settings, it can be observed that they do not achieve accuracy differences greater than 10, which occurs when the (*surrogate*, *attack*) pairs are (LR, PWWS) and (RF,

TF) respectively. In contrast, the worst strategy is IR, as the accuracy is only decreased by 3, regardless of the surrogate model adopted.

More insights can be obtained by analyzing the right part of Figure 12, where the *average* of both metrics for each attack is depicted. In terms of *Accuracy Difference*, the best attack is TF, which allowed a decrease of 9.3 on the target model; PWWS and TB showed similar results. The outcome confirms the findings already obtained (Figure 11) for the configurations (c) and (d), except for IR. In fact, it has, on average, a very high percentage of samples that fool the surrogate (i.e., successfully attack the surrogate), but a limited effectiveness in decreasing the accuracy of the target. This result can be attributed to the random nature of IR, which prioritizes random word deletions over targeted perturbations, unlike other attacks that strategically modify the text content to deceive the model.

The last analysis concerns the focus of the proposed work, i.e., the number of queries made on the target model. Recall that in a real-world scenario, making an innumerable number of queries to the remote system could activate defense mechanisms that could compromise the attacker's purposes. Table 11 summarizes the average number of queries for three of the configurations so long discussed in this section. For the **configurations (c)**, the highest and lowest number of queries is made by PWWS and IR, while they are TF and IR for the **configurations (d)**. As might be expected (also, referring to Figure 11), the techniques characterized by a higher number of queries are also those that are more likely to be successful in fooling the model, be it the surrogate or the target. The only exception is always IR, which, as always, shows a random behavior.

To understand the benefits of the proposed solution, these values must then be compared with those of our proposal. In particular, the comparison must consider the results of **configuration (c)** with those of the "Surrogate" column in **configuration (e)** and, likewise, the values of **configuration (d)** with the "Target" column in **configuration (e)**. In the first case, the findings from the comparison are not surprising since, on average, similar results are obtained as before: PWWS continues to be the technique with the most queries, and IR the worst. In the second case, however, it can be seen that the proposed approach performs on aver-

Table 11: Average number of queries for **configuration (c)**, **configuration (d)**, and **configuration (e)**. In the last case, the queries on the surrogate and target are considered separately.

	configuration (c)			**configuration (d)**	**configuration (e)**	
Attack	**LR**	**SVM**	**RF**	**NN**	**Surrogate**	**Target**
PWWS	**95.77**	**132.82**	**90.44**	79.36	**104.32**	**1**
TF	64.42	112.20	65.89	**113.92**	82.72	**1**
TB	39.22	64.99	40.08	39.70	49.11	**1**
IR	19.62	26.63	19.59	13.73	20.59	**1**
DWB	23.22	36.43	24.10	23.70	28.11	**1**

age only one query to the target as opposed to **configuration (d)**. This result is justified by the inherent nature of the proposed approach, which (i) starts from the original test set of n samples, (ii) tries to perturb each sample via the chosen attack strategy, and (iii) composes a new test set of n samples consisting of both the texts that did not fool the surrogate and those that were perturbed. Finally, the generated test set is evaluated on the target, and then a query is made for each sample.

6.4 Summary and Future Work

The introduction of a surrogate model situated between the attacker and the target model acts as a medium for efficiently transferring offline attacks in minimal queries, ensuring successful adversarial manipulation while minimizing resource consumption and potential disruption to operational systems. It should be acknowledged that the attacks encounter limitations, failing to perturb all the samples in our dataset. Consequently, the effectiveness of the targeted attack is reduced to a limited extent. As a general trend, the efficacy of the attack diminishes proportionally with the perturbed samples. Thus, for robust evaluations, a comprehensive perturbation coverage is required.

Chapter 7

Conclusion

In this thesis, several major limitations in the field of fake news detection were identified and addressed through the design of suitable solutions. Recognizing the lack of comprehensive datasets for fake news detection, we proposed a semi-automated methodology for the large-scale collection of news data, particularly from the platform X. This system enables the extraction of rich contextual and user-related information, resulting in a more complete dataset. To validate its utility, the dataset was evaluated using three unsupervised models, which yielded promising and balanced results.

Beyond introducing a comprehensive data collection framework, this thesis proposes a robust and effective ensemble learning-based architecture for fake news. By integrating multiple perspectives for news and successfully combining domain-specific knowledge, the system demonstrated superior performance in distinguishing false content and authentic news.

Additionally, a real-world propagation scenario was evaluated by incorporating user-centric factors that influence the decision to share news on online social networks. Specifically, friendship strength and mutual interests between users were considered as key factors in the dissemination process. The system achieved strong results, demonstrating its effectiveness not only in detecting fake news but also in containing its

further propagation within the network.

Finally, the literature review revealed that the traditional fake news detection models, primarily based on machine learning architectures, are highly prone to being attacked by adversaries. When the internal architecture of such models is known, adversarial attacks can be executed relatively easily. However, in a black-box scenario, where the internal functioning of the system remains unknown, the feasibility of attacks is affected. To overcome this limitation, we proposed the introduction of a surrogate model between the attacker and the fake news detection system. This approach not only reduced the computational burden and time associated with sending multiple queries to the target system but also bypassed the defense mechanism of the model, which is designed to prevent adversarial manipulation. As a result, a successful attack was possible, leading to a significant deterioration in the models' performance.

In conclusion, this thesis makes a significant contribution to the field of fake news detection by proposing a novel dataset construction methodology, a robust ensemble-based detection model integrating domain knowledge, an innovative method to detect and limit fake news in online social networks, and efficient adversarial attack strategies.

While some limitations remain, the presented studies provide a strong foundation for future research, which can build upon these findings to further enhance the reliability, robustness, and applicability of fake news detection systems.

Appendix A

Permissions

The material in Chapter 3 is reproduced from: F. Batool, G. Lo Re, and M. Morana, "Annotated Dataset Creation for Fake News Detection on Online Social Networks," AINA 2025, © Springer Nature. Included in the thesis with the publisher's permission for non-commercial academic use.

The content presented in chapter 4 is derived from: F. Batool, G. Lo Re, M. Morana, and M. Tortorici, "Multi-Domain Fake News Detection Exploiting Ensemble Learning Techniques," Joint National Conference on Cybersecurity (ITASEC & SERICS 2025).

The material in Chapter 5 is based on the author's previously published paper: F. Batool, G. Lo Re, M. Morana, and G. Rizzo, "Blocking Fake News Propagation exploiting OSNs Users Interests and Connections," 5th Intelligent Cybersecurity Conference (ICSC), 2025. IEEE, pp. 49–54.

And Chapter 6 is reproduced from F. Batool, F. Canino, G. Lo Re, and M. Morana, "A Black-box Adversarial Attack on Fake News Detection Systems," Italian Conference on CyberSecurity 2024.

Bibliography

Agarwal, Vasu et al. (2019). "Analysis of classifiers for fake news detection". In: *Procedia Computer Science* 165, pp. 377–383.

Aggarwal, Charu, Gewen He, and Peixiang Zhao (2016). "Edge classification in networks". In: *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, pp. 1038–1049.

Ali, Hassan et al. (2021). "All your fake detector are belong to us: evaluating adversarial robustness of fake-news detectors under black-box settings". In: *IEEE Access* 9, pp. 81678–81692.

Baarir, Nihel Fatima and Abdelhamid Djeffal (2021). "Fake News detection Using Machine Learning". In: *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH)*, pp. 125–130. DOI: 10.1109/IHSH51661.2021.9378748.

Bahad, Pritika, Preeti Saxena, and Raj Kamal (2019). "Fake news detection using bi-directional LSTM-recurrent neural network". In: *Procedia Computer Science* 165, pp. 74–82.

Barua, Zapan et al. (2020). "Effects of misinformation on COVID-19 individual responses and recommendations for resilience of disastrous consequences of misinformation". In: *Progress in Disaster Science* 8, p. 100119.

Batool, Farwa, Federico Canino, et al. (2022). "A Black-box Adversarial Attack on Fake News Detection Systems". In.

Batool, Farwa, Giuseppe Lo Re, and Marco Morana (2025). "Annotated Dataset Creation for Fake News Detection on Online Social Networks". In: *International Conference on Advanced Information Networking and Applications*. Springer, pp. 48–58.

Batool, Farwa, Giuseppe Lo Re, Marco Morana, and Giuseppe Rizzo (2025). "Blocking Fake News Propagation exploiting OSNs Users Interests

and Connections". In: *2025 5th Intelligent Cybersecurity Conference (ICSC)*. IEEE, pp. 49–54.

Batool, Farwa, Giuseppe Lo Re, Marco Morana, and Mario Tortorici (2025). "Multi-Domain Fake News Detection Exploiting Ensemble Learning Techniques". In.

Bhutani, Bhavika et al. (2019). "Fake news detection using sentiment analysis". In: *2019 twelfth international conference on contemporary computing (IC3)*. IEEE, pp. 1–5.

Biggio, Battista and Fabio Roli (2018). "Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning". In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. CCS '18. Toronto, Canada: Association for Computing Machinery, pp. 2154–2156. ISBN: 9781450356930.

Borgs, Christian et al. (2014). "Maximizing social influence in nearly optimal time". In: *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, pp. 946–957.

Budak, Ceren, Divyakant Agrawal, and Amr El Abbadi (2011). "Limiting the spread of misinformation in social networks". In: *Proceedings of the 20th International Conference on World Wide Web*. WWW '11. Hyderabad, India: Association for Computing Machinery, pp. 665–674. ISBN: 9781450306324. DOI: 10.1145/1963405.1963499.

Buskens, Vincent (2020). "Spreading information and developing trust in social networks to accelerate diffusion of innovations". In: *Trends in Food Science & Technology* 106, pp. 485–488. ISSN: 0924-2244.

Buzea, Marius Cristian, Stefan Trausan-Matu, and Traian Rebedea (2022). "Automatic fake news detection for romanian online news". In: *Information* 13.3, p. 151.

Chen, Duanbing et al. (2012). "Identifying influential nodes in complex networks". In: *Physica a: Statistical mechanics and its applications* 391.4, pp. 1777–1787.

Chen, Hao and YT Wang (2012). "Threshold-based heuristic algorithm for influence maximization". In: *Journal of Computer Research and Development* 49.10, pp. 2181–2188.

Chen, Mengyang et al. (2023). "Can large language models understand content and propagation for misinformation detection: An empirical study". In: *arXiv preprint arXiv:2311.12699*.

Chen, Wei, Yifei Yuan, and Li Zhang (2010). "Scalable influence maximization in social networks under the linear threshold model". In: *2010 IEEE international conference on data mining*. IEEE, pp. 88–97.

Cheng, Lu et al. (2021). "Causal Understanding of Fake News Dissemination on Social Media". In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. KDD '21. Virtual Event, Singapore, pp. 148–157. ISBN: 9781450383325.

Choudhury, Deepjyoti and Tapodhir Acharjee (2023). "A novel approach to fake news detection in social networks using genetic algorithm applying machine learning classifiers". In: *Multimedia Tools and Applications* 82.6, pp. 9029–9045.

Concone, Federico et al. (2024). "AdverSPAM: Adversarial SPam account manipulation in online social networks". In: *ACM Transactions on Privacy and Security* 27.2, pp. 1–31.

Cook, Wade D. (2006). "Distance-based and ad hoc consensus models in ordinal preference ranking". In: *European Journal of Operational Research* 172.2, pp. 369–385. ISSN: 0377-2217.

Du, Jiangshu et al. (2021). "Cross-lingual covid-19 fake news detection". In: *2021 international conference on data mining workshops (ICDMW)*. IEEE, pp. 859–862.

Emond, Edward J. and David W. Mason (2002). "A new rank correlation coefficient with application to the consensus ranking problem". In: *Journal of Multi-Criteria Decision Analysis* 11.1, pp. 17–28.

Ester, Martin et al. (1996). "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *kdd*. Vol. 96. 34, pp. 226–231.

Feng, Shi et al. (2018). "Pathologies of neural models make interpretations difficult". In: *arXiv preprint arXiv:1804.07781*.

Fink, Christian G et al. (2023). "A Congressional Twitter network dataset quantifying pairwise probability of influence". In: *Data in Brief*.

Fumera, Giorgio and Fabio Roli (2005). "A theoretical and experimental analysis of linear combiners for multiple classifier systems". In: *IEEE transactions on pattern analysis and machine intelligence* 27.6, pp. 942–956.

Gaglio, Salvatore, Andrea Giammanco, et al. (2021). "Adversarial machine learning in e-health: Attacking a smart prescription system". In: *International Conference of the Italian Association for Artificial Intelligence*. Springer, pp. 490–502.

Gaglio, Salvatore, Giuseppe Lo Re, and Marco Morana (2015). "Real-time detection of twitter social events from the user's perspective". In: *2015 IEEE International Conference on Communications (ICC)*, pp. 1207–1212. DOI: 10.1109/ICC.2015.7248487.

— (2016). "A framework for real-time Twitter data analysis". In: *Computer Communications* 73. Online Social Networks, pp. 236–242. ISSN: 0140-3664.

Gao, Ji et al. (2018). "Black-box generation of adversarial text sequences to evade deep learning classifiers". In: *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, pp. 50–56.

Goel, Pratyush et al. (2021). "Multi Domain Fake News Analysis using Transfer Learning". In: *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 1230–1237. DOI: 10.1109/ICCMC51019.2021.9418411.

Gong, Zaiwu et al. (2020). "Measuring trust in social networks based on linear uncertainty theory". In: *Information Sciences* 508, pp. 154–172. ISSN: 0020-0255.

Goyal, Amit, Wei Lu, and Laks VS Lakshmanan (2011). "Simpath: An efficient algorithm for influence maximization under the linear threshold model". In: *2011 IEEE 11th international conference on data mining*. IEEE, pp. 211–220.

Granovetter, Mark (1978). "Threshold Models of Collective Behavior". In: *American Journal of Sociology* 83.6, pp. 1420–1443. ISSN: 00029602, 15375390. (Visited on 03/21/2025).

Gravanis, Georgios et al. (2019). "Behind the cues: A benchmarking study for fake news detection". In: *Expert Systems with Applications* 128, pp. 201–213.

Guo, Weiwei et al. (2021). "Linear uncertain extensions of the minimum cost consensus model based on uncertain distance and consensus utility". In: *Information Fusion* 70, pp. 12–26. ISSN: 1566-2535.

Guo, Ying and Wei Song (2022). "A temporal-and-spatial flow based multimodal fake news detection by pooling and attention blocks". In: *IEEE Access* 10, pp. 131498–131508.

Han, Xiaoting and Li Niu (2013). "On charactering of information propagation in online social networks". In: *Journal of Networks* 8.1, p. 124.

Hangal, Sudheendra et al. (2010). "All friends are not equal: Using weights in social graphs to improve search". In: *Workshop on Social Network Mining & Analysis, ACM KDD*. Vol. 130.

Hutto, Clayton and Eric Gilbert (2014). "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In: *Proceedings of the international AAAI conference on web and social media*. Vol. 8. 1, pp. 216–225.

Jaho, Eva, Merkouris Karaliopoulos, and Ioannis Stavrakakis (2011). "ISCoDe: a framework for interest similarity-based community detec-

tion in social networks". In: *2011 IEEE conference on Computer communications workshops (INFOCOM WKSHPS)*. IEEE, pp. 912–917.

Jin, Di et al. (2020). "Is bert really robust? a strong baseline for natural language attack on text classification and entailment". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 05, pp. 8018–8025.

Jin, Zhiwei et al. (2017). "Multimodal fusion with recurrent neural networks for rumor detection on microblogs". In: *Proceedings of the 25th ACM international conference on Multimedia*, pp. 795–816.

Jung, Kyomin, Wooram Heo, and Wei Chen (2012). "Irie: Scalable and robust influence maximization in social networks". In: *2012 IEEE 12th international conference on data mining*. IEEE, pp. 918–923.

Karimi, Hamid et al. (2018). "Multi-source multi-class fake news detection". In: *Proceedings of the 27th international conference on computational linguistics*, pp. 1546–1557.

Kempe, David, Jon Kleinberg, and Éva Tardos (2003). "Maximizing the spread of influence through a social network". In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146.

Khanam, Zeba et al. (2021). "Fake news detection using machine learning approaches". In: *IOP conference series: materials science and engineering*. Vol. 1099. 1. IOP Publishing, p. 012040.

Kim, Jinha, Seung-Keol Kim, and Hwanjo Yu (2013). "Scalable and parallelizable processing of influence maximization for large-scale social networks?" In: *2013 IEEE 29th international conference on data engineering (ICDE)*. IEEE, pp. 266–277.

Koenders, Camille et al. (2021). "How Vulnerable Are Automatic Fake News Detection Methods to Adversarial Attacks?" In: *arXiv preprint arXiv:2107.07970*.

Koka, Sahas, Anthony Vuong, and Anish Kataria (2024). "Evaluating the efficacy of large language models in detecting fake news: a comparative analysis". In: *arXiv preprint arXiv:2406.06584*.

Kondamudi, Medeswara Rao et al. (2023). "A comprehensive survey of fake news in social networks: Attributes, features, and detection approaches". In: *Journal of King Saud University-Computer and Information Sciences* 35.6, p. 101571.

Konkobo, Pakindessama M et al. (2020). "A deep learning model for early detection of fake news on social media". In: *2020 7th International Conference on Behavioural and Social Computing (BESC)*. IEEE, pp. 1–6.

Kuhlman, Chris J. et al. (2013). "Blocking Simple and Complex Contagion by Edge Removal". In: *2013 IEEE 13th International Conference on Data Mining*, pp. 399–408. DOI: 10.1109/ICDM.2013.47.

Li, Jinfeng et al. (2018). "Textbugger: Generating adversarial text against real-world applications". In: *arXiv preprint arXiv:1812.05271*.

Liu, Bingsheng et al. (2019). "Large-scale group decision making model based on social network analysis: Trust relationship-based conflict detection and elimination". In: *European Journal of Operational Research* 275.2, pp. 737–754. ISSN: 0377-2217.

Liu, Ruifang et al. (2014). "Weighted graph clustering for community detection of large social networks". In: *Procedia Computer Science* 31, pp. 85–94.

Loeb, Stacy et al. (2020). "Fake news: spread of misinformation about urological conditions on social media". In: *European urology focus* 6.3, pp. 437–439.

Lyu, Shikun and Dan Chia-Tien Lo (2020). "Fake news detection by decision tree". In: *2020 SoutheastCon*. IEEE, pp. 1–2.

Mahara, Tripti et al. (2023). "Deep vs. shallow: a comparative study of machine learning and deep learning approaches for fake health news detection". In: *IEEE Access* 11, pp. 79330–79340.

Meyers, Marion, Gerhard Weiss, and Gerasimos Spanakis (2020). "Fake news detection on twitter using propagation structures". In: *Multidisciplinary International Symposium on Disinformation in Open Online Media*. Springer, pp. 138–158.

Milovanović, Stevan et al. (2019). "An approach to identify user preferences based on social network analysis". In: *Future Generation Computer Systems* 93, pp. 121–129. ISSN: 0167-739X.

Mohammad, Saif M and Peter D Turney (2013). "Nrc emotion lexicon". In: *National Research Council, Canada* 2, p. 234.

Nan, Qiong et al. (2021). "MDFEND: Multi-domain fake news detection". In: *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 3343–3347.

Ou, Chengeng et al. (2017). "Modelling heterogeneous information spreading abilities of social network ties". In: *Simulation Modelling Practice and Theory* 75, pp. 67–76.

Panda, Subhadarshi and Sarah Levitan (2022). "Deception detection within and across domains: Identifying and understanding the performance gap". In: *ACM Journal of Data and Information Quality* 15.1, pp. 1–27.

Papageorgiou, Eleftheria, Christos Chronis, et al. (2024). "A survey on the use of large language models (llms) in fake news". In: *Future Internet* 16.8, p. 298.

Papageorgiou, Eleftheria, Iraklis Varlamis, and Christos Chronis (2025). "Harnessing Large Language Models and Deep Neural Networks for Fake News Detection". In: *Information* 16.4, p. 297.

Papernot, Nicolas et al. (2017). "Practical Black-Box Attacks against Machine Learning". In: *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*. ASIA CCS '17. Abu Dhabi, United Arab Emirates: Association for Computing Machinery, pp. 506–519. ISBN: 9781450349444.

Patwa, Parth et al. (2021). "Fighting an infodemic: Covid-19 fake news dataset". In: *International Workshop on Combating Online Hostile Posts in Regional Languages during Emergency Situation*. Springer, pp. 21–29.

Qi, Peng et al. (2019). "Exploiting multi-domain visual information for fake news detection". In: *2019 IEEE international conference on data mining (ICDM)*. IEEE, pp. 518–527.

Rashkin, Hannah et al. (Sept. 2017). "Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-Checking". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 2931–2937.

Rasool, Tayyaba et al. (2019). "Multi-label fake news detection using multi-layered supervised learning". In: *Proceedings of the 2019 11th international conference on computer and automation engineering*, pp. 73–77.

Rath, Bhavtosh, Wei Gao, and Jaideep Srivastava (2019). "Evaluating vulnerability to fake news in social networks: A community health assessment model". In: *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pp. 432–435.

Raza, Shaina (Aug. 2021). "Automatic Fake News Detection in Political Platforms - A Transformer-based Approach". In: *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*. Ed. by Ali Hürriyetoğlu. Online: Association for Computational Linguistics, pp. 68–78. DOI: 10.18653/v1/2021.case-1.10. URL: https://aclanthology.org/2021.case-1.10/.

Raza, Shaina and Chen Ding (2022). "Fake news detection based on news content and social contexts: a transformer-based approach". In: *International Journal of Data Science and Analytics* 13.4, pp. 335–362.

Ren, Shuhuai et al. (2019). "Generating natural language adversarial examples through probability weighted word saliency". In: *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 1085–1097.

Rocha, Yasmim Mendes et al. (2023). "The impact of fake news on social media and its influence on health during the COVID-19 pandemic: A systematic review". In: *Journal of Public Health* 31.7, pp. 1007–1016.

Santia, Giovanni and Jake Williams (2018). "Buzzface: A news veracity dataset with facebook user commentary and egos". In: *Proceedings of the international AAAI conference on web and social media*. Vol. 12. 1, pp. 531–540.

Seddari, Noureddine et al. (2022). "A hybrid linguistic and knowledge-based analysis approach for fake news detection on social media". In: *IEEE Access* 10, pp. 62097–62109.

Shakarian, Paulo et al. (2015). "The Independent Cascade and Linear Threshold Models". In: *Diffusion in Social Networks*. Cham: Springer International Publishing, pp. 35–48. ISBN: 978-3-319-23105-1.

Shao, Huajie, Dachun Sun, et al. (2020). "Truth discovery with multi-modal data in social sensing". In: *IEEE Transactions on Computers* 70.9, pp. 1325–1337.

Shao, Huajie, Shuochao Yao, et al. (2018). "A constrained maximum likelihood estimator for unguided social sensing". In: *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, pp. 2429–2437.

Shelke, Sushila and Vahida Attar (2019). "Source detection of rumor in social network–a review". In: *Online Social Networks and Media* 9, pp. 30–42.

Shibata, Yusuxke et al. (1999). "Byte pair encoding: A text compression scheme that accelerates pattern matching". In.

Shrivastava, Priya and Dilip Kumar Sharma (2022). "Covid-19 fake news detection using pre-tuned bert-based transfer learning models". In: *2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART)*. IEEE, pp. 64–68.

Shu, Kai, Deepak Mahudeswaran, et al. (2020). "Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media". In: *Big data* 8.3, pp. 171–188.

Shu, Kai, Amy Sliva, et al. (2017). "Fake news detection on social media: A data mining perspective". In: *ACM SIGKDD explorations newsletter* 19.1, pp. 22–36.

Silva, Amila et al. (2021). "Embracing domain differences in fake news: Cross-domain fake news detection using multi-modal data". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 35. 1, pp. 557–565.

Sitaula, Niraj et al. (2020). "Credibility-based fake news detection". In: *Disinformation, misinformation, and fake news in social media: Emerging research challenges and Opportunities*. Springer, pp. 163–182.

Steenari, Jussi and Jukka K Nurminen (2023). "Geospatial DBSCAN Hyperparameter Optimization with a Novel Genetic Algorithm Method". In.

Subramani, Kumar et al. (2011). "Density-based community detection in social networks". In: *2011 IEEE 5th International Conference on Internet Multimedia Systems Architecture and Application*. IEEE, pp. 1–8.

Sun, Yanwen et al. (2024). "MDFM: A Multi-Domain Fake News Detection Method Fusing Memory Features". In: *2024 9th International Conference on Computer and Communication Systems (ICCCS)*, pp. 1272–1277. DOI: 10.1109/ICCCS61882.2024.10602908.

Tang, Youze, Xiaokui Xiao, and Yanchen Shi (2014). "Influence maximization: Near-optimal time complexity meets practical efficiency". In: *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pp. 75–86.

Teo, Ting Wei et al. (2024). "Integrating large language models and machine learning for fake news detection". In: *2024 20th IEEE International Colloquium on Signal Processing & Its Applications (CSPA)*. IEEE, pp. 102–107.

Tufchi, Shivani, Ashima Yadav, and Tanveer Ahmed (2023). "A comprehensive survey of multimodal fake news detection techniques: advances, challenges, and opportunities". In: *International Journal of Multimedia Information Retrieval* 12.2, p. 28.

Verma, Pawan Kumar et al. (2021). "WELFake: word embedding over linguistic features for fake news detection". In: *IEEE Transactions on Computational Social Systems* 8.4, pp. 881–893.

Vijayaraghavan, Sairamvinay et al. (2020). "Fake news detection with different models". In: *arXiv preprint arXiv:2003.04978*.

Vishwakarma, Dinesh Kumar et al. (2023). "A framework of fake news detection on web platform using ConvNet". In: *Social Network Analysis and Mining* 13.1, p. 24.

Wang, Daokang et al. (2023). "Soft-Label for Multi-Domain Fake News Detection". In: *IEEE Access* 11, pp. 98596–98606. DOI: 10.1109/ACCESS.2023.3313602.

Wang, Dong et al. (2012). "On truth discovery in social sensing: A maximum likelihood estimation approach". In: *Proceedings of the 11th International Conference on Information Processing in Sensor Networks*, pp. 233–244.

Wang, Lanjun et al. (2024). "Bots shield fake news: adversarial attack on user engagement based fake news detection". In: *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 2369–2378.

Wardoyo, Retantyo et al. (2020). "Weighted majority voting by statistical performance analysis on ensemble multiclassifier". In: *2020 Fifth International Conference on Informatics and Computing (ICIC)*. IEEE, pp. 1–8.

Watts, Duncan J. (2002). "A simple model of global cascades on random networks". In: *Proceedings of the National Academy of Sciences of the United States of America* 99, pp. 5766–5771.

Xu, Yanxin et al. (2021). "Trust propagation and trust network evaluation in social networks based on uncertainty theory". In: *Knowledge-Based Systems* 234, p. 107610. ISSN: 0950-7051.

Yang, Peng and Peilin Zhao (2015). "A min-max optimization framework for online graph classification". In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 643–652.

Yang, Xuankai et al. (2024). "UPDATE: mining user-news engagement patterns for dual-target cross-domain fake news detection". In: *2024 IEEE 11th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, pp. 1–10.

Yang, Yuting et al. (2019). "How to write high-quality news on social network? predicting news quality by mining writing style". In: *arXiv preprint arXiv:1902.00750*.

Yu, Wencheng et al. (2022). "Multi-domain fake news detection for history news environment perception". In: *2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, pp. 428–433.

Zareie, Ahmad and Rizos Sakellariou (2022). "Rumour spread minimization in social networks: A source-ignorant approach". In: *Online Social Networks and Media* 29, p. 100206.

Zareie, Ahmad, Amir Sheikhahmadi, and Adel Fatemi (2017). "Influential nodes ranking in complex networks: An entropy-based approach". In: *Chaos, Solitons & Fractals* 104, pp. 485–494.

Zhang, Xichen and Ali A Ghorbani (2020). "An overview of online fake news: Characterization, detection, and discussion". In: *Information Processing & Management* 57.2, p. 102025.

Zhou, Xinyi and Reza Zafarani (2019). "Network-based fake news detection: A pattern-driven approach". In: *ACM SIGKDD explorations newsletter* 21.2, pp. 48–60.

Zhou, Zhixuan et al. (2019). "Fake news detection via NLP is vulnerable to adversarial attacks". In: *arXiv preprint arXiv:1901.09657*.

Zhu, Yongchun et al. (2022). "Memory-guided multi-view multi-domain fake news detection". In: *IEEE Transactions on Knowledge and Data Engineering*.

Zubiaga, Arkaitz et al. (2016). *PHEME Rumour Scheme Dataset: Journalism Use Case*. Version v2. figshare. DOI: 10.6084/m9.figshare.2068650.v2. URL: https://doi.org/10.6084/m9.figshare.2068650.v2.