DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master′s Thesis

Explainability of Fake News Detection Models for Social Media

Batuhan Erdogdu

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis

Explainability of Fake News Detection Models for Social Media

Erklärbarkeit von Modellen zur Fake-News-Erkennung in Sozialen Medien

Author:	Batuhan Erdogdu
Supervisor:	Prof. Dr. Georg Groh
Advisor:	M.Sc. Carolin Schuster
Submission Date:	Submission date

I confirm that this master's thesis is my own work and I have documented all sources and material used.

Munich, Submission date　　　　Batuhan Erdogdu

Acknowledgments

Abstract

Contents

1 Introduction

With the rapid development of communication technologies, social media has become one of the most frequently used news sources. It is easier, faster, and offers interaction with people. For example, a study from Pew Research Center (**NewsConsumptionAcrossSocialMedia_pe** reports that in 2021, 48% of U.S. adults got their news from social media "often" or “sometimes”. Furthermore, global data from 2022 (**StatistaUsageOfSocialMedia_Watson**) shows that over 70% of adults from Kenya, Malaysia, Phillippines, Bulgaria, and Greece use social media as one of their news sources, while this share is lower than 40% for the adults in the United Kingdom, The Netherlands, Germany, and Japan. These examples show that a considerable percentage of the population uses social media as a news source.

In contrast to its convenience, interactivity, and speed, social media can spread any kind of information since no regulatory authority checks the posts. As a result, a flood of false and misleading information is observed on social media (**SocialMediaAndFakeNewsIn2016Election_Allc**

The research community introduced numerous approaches to counteract the uncontrolled dissemination of fake news. For instance, some studies focused on building datasets (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**; **LiarLiarPantsOnFire_Wa** **FakeReddit_Nakamura**; **SomeLikeItHoaxDataset_Tacchini**; **BuzzfaceDataset_Santia**; **UPFD_Dataset_Shu**), and some studies leveraged the power of *Machine Learning* (ML) to automatically detect fake news (**FakeNewsDetectionUsingGeometricDeepLearning_Monti**; **GraphNeuralNetworksWithContinualLearningFakeNewsDetection_Han**; **RumorDetectionBidirection** **SAFEFND_Zhou**) by learning features from the data. Due to the number of posts and the limitation of staff to check the posts, ML-based techniques can reduce manual labor when used with human supervision to counter the spreading of fake news. However, ML-based techniques with high complexity, such as *Deep Neural Networks* (DNNs), are harder to understand and interpret since they act like black-boxes (**CanWeOpenTheBlackBoxOfAI_Castelvecchi**).

The integration of ML-based methods into human society impacts more people every day. While incredibly helpful in some aspects, ML-based techniques do not offer a reason for a particular prediction. Furthermore, we can not simply accept classification accuracy as a metric to evaluate real-world problems (**TowardsARigorousScienceML_Velez**). Integrating ML-based methods into human society makes interpretability a requirement to increase social acceptance (**InterpretableMachineLearning_Molnar**).

Consequently, a new research field called *eXplainable Artificial Intelligence* (XAI) surfaced to fill this missing link between humans and *Artificial Intelligence* (AI). XAI proposes creating a set of ML techniques that deliver more explainable models while preserving learning performance, and help humans to understand, properly trust, and effectively handle the emerging generation of artificially intelligent partners (**XAI_Gunning**). While incorporating XAI increases social acceptance, it also aims to create more privacy-aware (**SlaveToTheAlgorithm_EdwardsVeale**), fairer, and trustworthy systems (**TheMythosOfModelInterpretability_Lipton**).

Like all ML techniques, *Fake News Detection* (FND) models need interpretability, particularly when implementing countermeasures for fake news. However, the interpretability of a model is not often considered despite the large amount of research produced in the last decade. Incorporating social context (**FakeNewsNet_Shu**), representing the propagation networks as graphs (**UPFD_Dataset_Shu**), and using *Graph Neural Networks* (GNNs) to produce *State Of The Art* (SOTA) models (**FakeNewsDetectionUsingGeometricDeepLearning_** have increased the complexity, but also the performance of FND models. For instance, using social context data alone has proved to be more effective than textual data alone in recent studies (**UPFD_Dataset_Shu**). However, it is not clear which social features impact the decision process of these models.

This thesis focuses on the explainability of FND models using tools from the XAI suite. Specifically, we focus on content-based models and social context-based models to elaborate on their interpretability. Thus, we define three research objectives:

RO1 Determine the interpretation tools for explaining FND models.

RO2 Show that interpretations of FND models play an essential role in understanding the shortcomings of the FND models.

RO3 Determine which features impact the outcome the most.

In the next section, we elaborate on fake news, FND methods and xAI. We give foundations of fake news and define its characteristics. Then we categorize FND models and give important examples from literature. After examining fake news and detection methods for it, we focus on characterization of xAI, give definitions that will be used throughout this thesis.

In the third section, we examine FND models that were used in this thesis, and characterize them as defined in **??**. Furthermore, we deliver information about the model architecture, technologies and datasets used, and a detailed mathematical background on DNNs and GNNs. We also draw attention to some crucial matters such as model aging.

In the fourth section, we focus on xAI techniques that are used in this thesis. We give a comprehensive explanations and show their importance when dealing with complex

models. We illustrate results from our experiments, draw attention to shortcomings of models, and show a model is fair or not. We also discuss the plausability of the produced explanantions in this section.
In the last section, we talk about our overall findings. We show that research objectives are satisfied. Additionally, we discuss the limitations that we have encountered, and possible future works.

2 Background and Related Work

We explain two research fields that create the bedrock of this thesis, namely, fake news detection and explainable artificial intelligence. Both areas provide the foundation of tools used in this work. The first provides the mechanisms and approaches to detect fake news, and the second offers a suite of techniques to interpret these mechanisms and strategies.

Initially, in **??**, we discuss societal challenges, the characteristics, and the history of fake news. Then we talk about the detection methods that were developed over the years. After showing the challenges of creating FND models, we conclude the first section with SOTA FND models.

After fake news detection, in **??**, we first examine when XAI is necessary and its importance. Then, we define the suite of explainable artificial intelligence and the goals of XAI, and finally, we determine the suite that aims to satisfy these goals.

2.1 Fake News Detection

In the past decade, social media has become a place where anyone can share information. Although fast, free, and easy to access, obtaining real news from social media can be difficult, and one should do so at their own risk and always check the facts (**SocialMediaAndFakeNewsIn2016Election_Allcott**; **TheScienceOfFakeNews_Lazer**). Nevertheless, the news stream never ends; thus, the need to verify the credibility of news using automated systems arises. To address this necessity, the number of studies involving *Fake News* or *Fake News Detection* has dramatically increased in the last decade (Fig. **??**).

In **??**, we briefly present the history of fake news and look at studies that display the impact of fake news on society. In this section, we also define the terms fake news, disinformation, and misinformation.

In **??**, we make an excursion into social sciences and human psychology, delivering insights into why humans fall for or tend to believe fake news. Furthermore, we draw some insights from the social, technical, and data-oriented foundations of fake news.

We then list the available datasets used in FND and deliberate their advantages and disadvantages in **??**. Finally, in **??**, we summarize the evolution of detection algorithms,

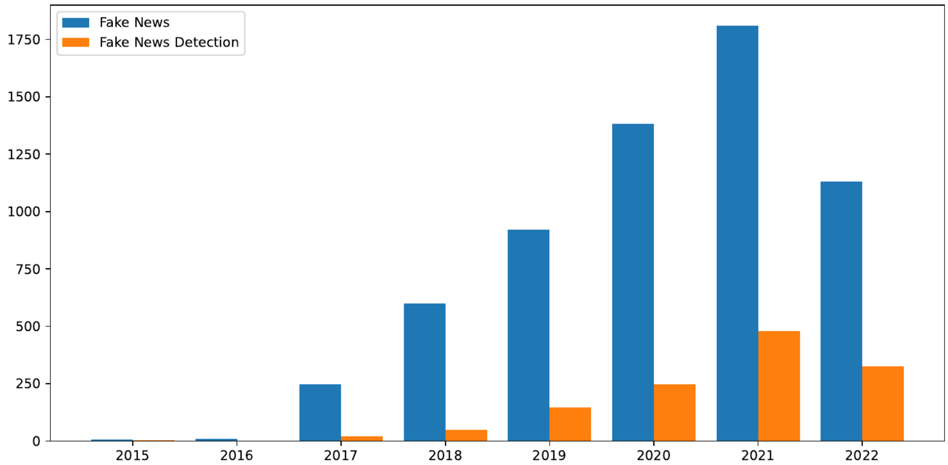

Figure 2.1: Total number of publications that include (1) *Fake News* (blue) and (2) *Fake News Detection* (orange) publications by year. Source: Scopus; Search Arguments: (1) TITLE-ABS-KEY("fake news*") PUBYEAR AFT 2014 (2) TITLE-ABS-KEY("fake news detection")

then we classify FND algorithms with respect to their input data type and what they focus on that data.

2.1.1 Fake News

Throughout history, various forms of widespread fake news have been recorded. For instance, in the thirteenth century BC, Rameses the Great decorated his temples with paintings that tell stories of victory in the Battle of Kadesh. However, the treaty between the two sides reveals that the battle's outcome was a stalemate (**HistorysGreatestLies_Weir**). Just after the printing press was invented in 1439, the circulation of fake news began. One of history's most famous examples of fake news is the "Great Moon Hoax" (**TheGreatMoonHoax_Foster**). In 1835, The Sun newspaper of New York published articles about a real-life astronomer and a made-up colleague who had observed life on the moon. It turns out that these fictionalized articles brought them new customers and almost no backlash after the newspaper admitted that the articles mentioned earlier were a hoax[1].
In order to highlight the difference, using the definitions from (**ThePsycologyOfFakeNews_Pennycook**), we formally introduce the terms disinformation and misinformation as follows.

[1] https://www.politico.com/magazine/story/2016/12/fake-news-history-long-violent-214535/

Definition 2.1.1 (*Disinformation*)**.** "*Information that is false or inaccurate and was created with a deliberate intention to mislead people.*" (**ThePsycologyOfFakeNews_Pennycook**)

Definition 2.1.2 (*Misinformation*)**.** "*Information that is false, inaccurate, or misleading. Unlike disinformation, misinformation does not necessarily need to be created deliberately to mislead.*" (**ThePsycologyOfFakeNews_Pennycook**)

There is no fixed definition for fake news. Thus, we elaborate on the definitions of fake news. A limited definition is news articles that are intentionally or verifiably false (**SocialMediaAndFakeNewsIn2016Election_Allcott**). This definition stresses authenticity and intent. The inclusion of false information that can be confirmed refers to authenticity. On the other hand, intent refers to the deceitful intention to delude news consumers (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**). This definition is widely used in other studies (**AutomaticDeceptionDetection_Conroy**; **TheFakeNewsSpreadingPlague_Mustafaraj**; **FakeNewsDetectionOnSocialMediaADataMiningPerspec**
Furthermore, recent social sciences studies (**TheScienceOfFakeNews_Lazer**; **ThePsycologyOfFakeNews_**
define fake news as fabricated information that mimics news media content in form but not in organizational process or intent. Similarly, this definition covers authenticity and intent; additionally, it includes the organizational process. More general definitions for fake news consider satire news as fake news due to the inclusion of false information even though satire news aim to entertain and inherently reveals its deception to the consumer (**WhenFakeNewsBecomesReal_Balmas**; **TheImpactOfRealNewsAboutFakeNews_Brewer**; **NewsVerificationByExploitingConflictingSocialVie**
FakeNewsOrTruthUsingSatiricalCues_Rubin). Further definitions include hoaxes, satires, and obvious fabrications (**DeceptionDetectionForFakeNews3TypesOfFakeNews_Rubin**)

In this thesis, we are not interested in the organizational process and do not consider conspiracy theories (**ConspiracyTheories_Sunstein**), superstitions (**Superstition_Lindeman**), rumors (**RumorsAndHealthCareReform_Berinsky**), misinformation, satire, or hoaxes. Therefore, we use the limited definition from (**SocialMediaAndFakeNewsIn2016Election_Allcott**) and formally introduce it.

Definition 2.1.3 (*Fake News*)**.** "*News articles that are intentionally or verifiably false.*" (**SocialMediaAndFakeNe**

Fake news can lead to disastrous situations, such as crashes in stock markets, resulting in millions of dollars. For example, Dow Jones industrial average went down like a bullet (see Fig. **??**) after a tweet about an explosion injuring President Obama went out due to a hack (**MarketQuaversAfterFakeAPTweet_ElBoghdady**).

The detrimental impacts of fake news further extend to societal issues. When fake news rose to prominence with the 2016 U.S. Presidential Election (**USPresidentialElection2016**), a man, convinced by what he read on social media about a pizzeria trafficking humans,

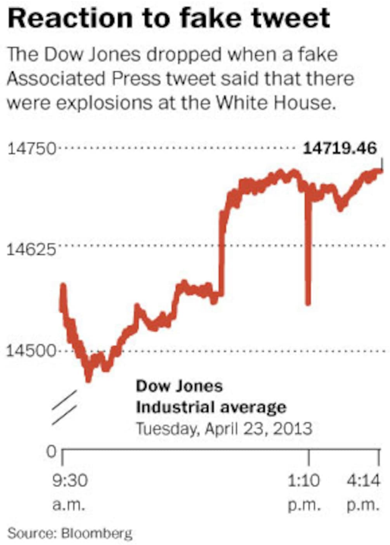

Figure 2.2: The market's reaction to the fake tweet. The sharp decline caused by a single tweet. Image obtained from (**MarketQuaversAfterFakeAPTweet_ElBoghdady**)

went on a shooting spree in that pizzeria. Later named Pizzagate (**Pizzagate_Fisher**), this incident illustrates the deadly impact of fake news. In fact, fake news can even affect presidential elections (**SocialMediaAndFakeNewsIn2016Election_Allcott**; **TrumpWonBecauseOfFacebook_Read**).
Recent history exhibits that some fake news spreads like wildfires through social media. Evidence shows that the most popular fake news stories were more widely shared than the most popular mainstream news stories (**Buzzfeed_FakeNewsOutperformRealNews_Silverman**). Digital News Report 2022 (**ReutersInstituteDigitalNewsReport**) reports in its key findings that trust in the news is 42% globally, the highest (69%) in Finland, and the lowest (26%) in the U.S.A. Additionally, the same study shows that in early 2022, in the week of the survey, between 45% and 55% of the surveyed social media consumers worldwide witnessed false or misleading information about COVID-19. The same study also reports the appearance of fake news in politics was between 34% and 51%, and between 9% and 48% for fake news about celebrities, global warming, and immigration (**StatistaUsageOfSocialMedia_Watson**).

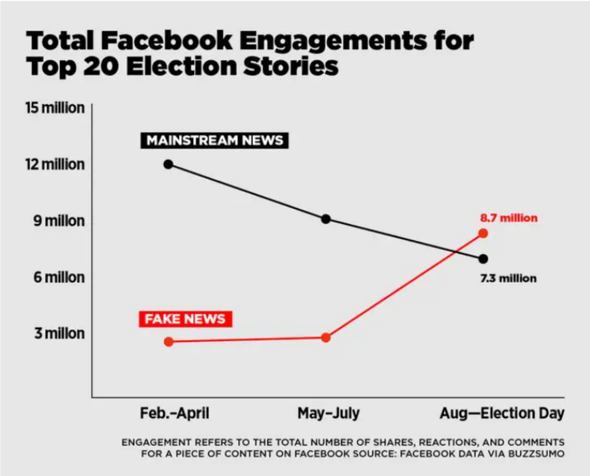

Figure 2.3: The rising engagement for fake news stories observed after May-July, just before Presidential Elections. Image obtained from (**Buzzfeed_FakeNewsOutperformRealNews_Silverman**)

2.1.2 Foundations of Fake News

The environment for fake news has been the traditional news media for a long time. First started with newsprint, then continued with radio and television, and now with social media and the web, the dissemination of fake news reached its peak. Next, we discuss the psychological and social foundations of fake news to stress the importance of human psychology, especially when accepting fake news as genuine and sharing it with others. Then we focus on the technical foundations where we discuss how social media and technologiy have accelerated the diffusion of fake news.

Psychological Foundations. Understanding the difference between real and fake news is not an easy task for a human. Two psychological theories, namely, *naive realism* and *confirmation bias*, examine why humans fall for fake news. The first refers to a person's disposition to believe that their point of view is the mere accurate one, while people who believe otherwise are uninformed or biased (**NaiveRealism_Reed**). The second, often called selective exposure, is the proclivity to prefer information that confirms existing views (**ConfirmationBias_Nickerson**).

Another reason for human fallacy in fake news is that once a misperception is formed, it becomes difficult to correct. In fact, it turns out that correcting people leads them to believe false information more, especially when given factual information that refutes their beliefs (**WhenCorrectionsFail_Nyhan**).

Social Foundations. The prospect theory explains the human decision-making process as a mechanism based on maximizing relative gains and minimizing losses with respect to the current state (**ProspectTheory_Kahneman**; **AdvancesInProspectTheory_Kahneman**). This inherent inclination to get the highest reward also applies to social cases in which a person will seek social networks that provide them with social acceptance. Consequently, people with different views tend to form separate groups, which makes them feel safer, leading to the consumption and dissemination of information that agrees with their opinions. These behaviors are explained by social identity theory (**SocialIdentityTheory_Ashforth**) and normative social influence (**NormativeSocialInfluence_Asch**). Two psychological factors play a crucial role here (**TheRussianFirehoseOfFalsehood_Paul**). The first, social credibility, is explained by a person's tendency to recognize a source as credible when that source is deemed credible by other people. The second, called the frequency heuristic, is the acceptance of a news piece by repetitively being exposed to it. Collectively, these psychological phenomena are closely related to the well-known filter bubble (**TheFilterBubble_Pariser**), also called echo chamber, which is the formation of homogenous bubbles in which the users are people of similar ideologies and share similar ideas. Being isolated from different views, these users usually are inclined to have highly polarized opinions (**EchoChambers_Sunstein**). As a result, the main reason for misinformation dispersal turned out to be the echo chambers (**TheSpreadingOfMisinformationOnline_DelVicario**).

Technical Foundations. Social media's easy-to-use and connected nature give rise to more people selecting or even creating their own news source. Naturally, this gives way to more junk information echoing in a group of people on social media. As algorithms evolve to understand user preferences, social media platforms recommend similar people or groups to those in echo chambers. A recent study (**TheEffectOfPeopleRecommenderOnEchoCham** shows that these recommenders can strengthen these echo chambers. They discuss that some of these recommenders contribute to the polarization on social media. In other words, people can convince themselves that any fake news is real by staying in their echo chambers. One main reason that some fake news spreads so rapidly on social media is the existence of malicious accounts. The account user can be an actual human or a social bot since creating accounts on social media is no cost and almost no effort. While many social bots provide valuable services, some were designed to harm, mislead, exploit, and manipulate social media discourse. Formally, a social bot is a social media account governed by an algorithm to fabricate content and interact with other users (**TheRiseOfSocialBots_Ferrara**). A more recent study from the same author shows that malicious social bots were heavily used in the 2016 U.S. Presidential Elections (**SocialBotsDistortThe2016USPresidentialElection_Bessi**). On the other hand, malicious accounts that are not bots, such as online trolls who aim to trigger negative emotions and humans that provoke people on social media to get an emotional response, contribute to the proliferation of fake news (**AnyoneCanBecomeATroll_Cheng**).

Building upon three foundations, we draw some results for fake news to be considered when building a fake news detection model:

1. *Invasive*: Fake news can appear on anyone's feed if it spreads for a sufficient amount of time.
2. *Hard to discern*: Fake news is fabricated in such a way that it resembles the authenticity of a real news source. This indistinguishability leads to issues when working with news-content-based FND models.
3. *The source is crucial*: The credibility of a news source is essential. We can use news from credible sources to teach the model to distinguish genuine from fabricated.
4. *Fake news has hot spots*: The echo chambers are invaluable examples when trying to understand the behaviors of fake news. We can leverage this attribute and use social models, such as graphs, to successfully detect fake news.
5. *Early detection is essential*: As discussed in psychological foundations, the volume of exposure to a piece of fake news can significantly affect one's opinions, thus leading to more misinformed individuals.

Data-Oriented Foundations. We define features for news content and social context to represent the news pieces in a structured manner. First, we introduce attributes for news content (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**):

- *Source*: Publisher of the news piece.
- *Headline*: Short title text that aims to catch the readers' attention and describes the article's main topic.
- *Body Text*: The main text piece that details the news story.
- *Image/Video*: Part of the body content supplies visual input to articulate the story.

Using these attributes, we extract two types of features for news content:

Linguistic-based features: The news content is heavily based on textual content. Thus, the first feature that belongs to this class is lexical features which make use of character and word level frequency information which can be obtained by the utilization of *term frequency-inverse term frequency* (TF-IDF) (**TF_Luhn**; **IDF_Jones**) or bag-of-words (BoW). The second feature is based on syntactic features which include sentence-level features that can be obtained via *n-grams* and punctuation and *parts-of-speech* (POS) (**POS_Daelemans**) tagging. We can extend these features to domain-specific ones, such as external links and the number of graphs (**AStylometricInquiry_Potthast**).

Visual-based features: Particularly for fake news, the visual content is a strong tool for establishing belief (**VisualMisAndDisinformation_Viorela**). Hence, the features that reside in images and videos become significant. Fake images and videos which brings the fake story together are commonly used(e.g. **PutinBehindBars_Harding**; **DeepFakeQueensSpeech_Sawer**). To counteract the effects of misleading visual input, recent studies (**ExploitingMultiDomainVisualInformation_Qi**) examined visual and statistical information for fake news detection. Visual features consist of clarity score, similarity distribution histogram, diversity score, and clustering score. Statistical features are listed as count, image ratio, multi-image ratio etc. (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**).

Now, we define features for social context, which has recently drawn much attention from the research community (**BeyondNewsContents_Shu**; **HierarchicalPropagationNetworksForFND_S** Overall, we will concern three aspects of social context data: user-based, post-based, and network-based features.

User-based: As mentioned in the Technical Foundations part of this subsection, fake news has various ways of disseminating, such as via echo chambers, malicious accounts, or bots. Therefore, analyzing user-based information can prove useful. We distinguish user-based features at the group and individual levels (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**). Individual levels are extracted to deduce the credibility of each user by utilizing, for example, the number of followers and followees, the number of tweets authored by a user, etc (**InformationCredibilityOnTwiter_Castillo**). On the other hand, group-level user-based features are the general characteristics of groups of users related to the news (**AutomaticDetectionOfRumor_Yang**). Parallel to the social identity theory and normative social influence idea, the assumption is the consumers of real and fake news tend to form different groups, which may lead to unique characteristics. Typical group-level features stem from individual-level features by obtaining the share of verified users, and the average number of followers and followees (**DetectRumorsUsingTimeSeries_Ma**).

Post-based: Analysis of reactions by users can prove helpful when determining whether a news piece is real or not. For example, if a news piece is getting doubtful comments, this can help determine the news piece's credibility. As such, post-based features are based on inferring the integrity of a news piece from three levels. Namely, post-level, group-level, and temporal-level (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**). Post-level features can be embedding values for each post or take forms as mentioned in linguistic-based features, e.g., n-grams, BoW, etc. For post-level features, we can also consider general approaches such as topic extraction (e.g., using latent Dirichlet allocation (LDA) (**LatentDirichletAllocation_Blei**)), stance extraction, which provides information about users' opinions (e.g., supports, opposes (**NewsVerificationByExploitingConflictingSocialViewpoints_Jin**)), and finally credibility extraction, which deals with estimating the degree of trust for each post (**InformationCredibilityOnTwiter_Castillo**). Group-level post-based features collect feature values for all relevant posts and apply an operation to extract pooled information. When determining the credibility of news, group-level features proved to be helpful (**NewsVerificationByExploitingConflictingSocialViewpoints_Jin**). Temporal-level features deal with changes in post-level features over time. Typically, unsupervised learning methods such as Recurrent Neural Networks (RNN) are employed to capture the changes over time (**DetectingRumorsFromMicroblogs_Ma**).

Network-based: As discussed in the Technical Foundations part, fake news is likely to give rise to echo chambers, which leads to the idea of a network-based approach. When represented as networks, the propagation behavior of fake news can be analyzed further, and patterns can be discovered (**FakeNewsDetectionOnSocialMediaADataMiningPe**
In literature, various types of networks exist, the most common ones are stance networks, occurrence networks, and friendship networks. Stance networks are constructed upon stance detections which is a part of sentiment analysis and deal with determining a user's viewpoint using text and social data (**StanceClassificationAttention_Du**). Using all users' stances, a network is built in which the nodes are the tweets relevant to the news piece and the edges represent the similarity of stances between nodes (**NewsVerificationByExploitingConflictingSocialViewpoints_Jin**; **SomeLikeItHoaxDataset_Tacchini**). On the other hand, occurrence networks leverage the frequency of mentions or replies about the same news piece (**ProminentFeaturesOfRum**
Friendship networks are based on the follower/followee relationship of users who share posts connected to the news piece. Derived from friendship networks, in the form of one of the datasets we use in our experiments (**UPFD_Dataset_Shu**), diffusion networks are designed to track the course of the dissemination of news (**ProminentFeaturesOfRumorPropagation_Kwon**). Briefly, a diffusion network consists of nodes representing users and diffusion paths representing the relationship and interaction between users. In detail, a diffusion path between two users u_i and u_j exists if and only if u_j follows u_i, and u_j shares a post about a news piece that u_i has already shared a post about (**FakeNewsDetectionOnSocialMediaADataMiningPer**
It has been shown that characterizing these networks is possible (**ProminentFeaturesOfRumorPropag**
Approaches for these networks have gained traction recently, especially with some SOTA GNNs, e.g., (**FakeNewsDetectionUsingGeometricDeepLearning_Monti**).

To conclude this subsection, we have covered psychological, social, technical, and data-oriented foundations in this section. We established that, from different aspects, there are various reasons for the dissemination of fake news. Accordingly, we consider these reasons when building FND systems. In the next section, we discuss FND approaches and how they have evolved. Moreover, we characterize FND models and talk about each type of approach.

2.1.3 Evolution of Fake News Detection

Fake news detection is as old as fake news itself. Before social media became a hub for news consumers, fact-checkers, i.e., fake news detectors, were only journalists and literate people. Following the source shift of the news from printed paper to online, then social media, detection of fabricated news have become costly, cumbersome, and not as rewarding due to the endless stream of information and decreasing trust in journalism. Automatic detection for news thus became a necessity in our world (**NewsInAnOnlineWorld_Chen**).
Similar to what we did in the Data-Oriented Foundations part of the previous subsection, we classify fake news detection models as *News Content Models* and *Social Context Models* (see **??**) and start with News Content Models by following the classification principles in (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**).

News Content Models. Based on news content and fact-checking methodologies, these models are the starting point of fake news detection. News content models are classified as Knowledge-based and Style-based. We first introduce style-based models as they are the initial approaches for FND.

Style-based: Previous research in psychology has mainly focused on style-based approaches to detect *manipulators* in the text. Particularly deception detection techniques were popular and commonly developed in early works in criminology and psychology. We describe two different ways to approach style-based news content models, namely, *Deception-oriented* and *Objectivity-oriented* (**FakeNewsDetectionOnSocialMediaADa**

- *Deception-oriented*: The initial approaches for automated fake news detection focus on news context and stem from deception detection in language. The first study that focuses deception detection in language (**DieEntwicklungDerGerichtspsycholog** hypothesized that the truthfulness of the statement is more important than the integrity of the reporting person, and there exist definable and descriptive criteria that form a crucial mechanism for the determination of the truthfulness of statements. Even though this study is from experimental psychology, it stresses the feasibility of defining a set of rules that determine the truthfulness of a statement.

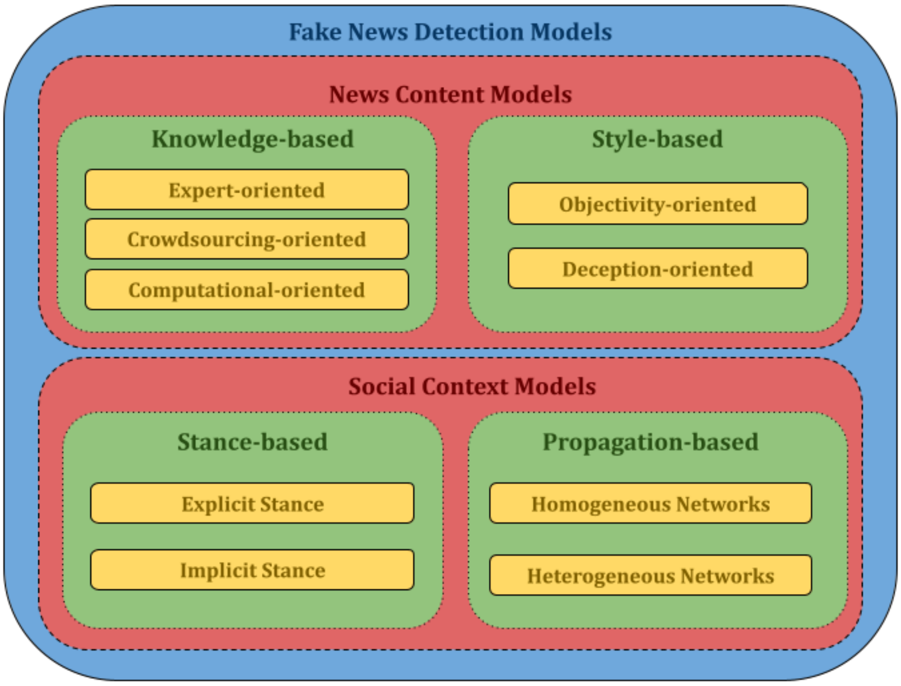

Figure 2.4: Characterization of Fake News Detection Models, Figure inspired by Figure 1 in **FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**.

An early study from criminology, Scientific Content Analysis (SCAN) (**SCAN_Sapir1987**), analyzes freely written statements. In this process, SCAN claims to detect potential instances of deception in the text but cannot label a statement as a lie or truth. The next study for SCAN (**SCAN_Smith2001**) is the first known study that correlates linguistic features with deceptive behavior using high-stakes data. Similar to SCAN, the subsequent studies (**CommunicationUnderStress_Adams**; **LyingWords_Newman**) that link linguistic features to deception classify the owner of the statement as truth-teller or liar according to the frequency of deception indicators in the statement.
Although for automated deception detection, defining a methodology is more challenging (**TheAccuracyConfidenceRelation_DePaulo**), early studies have shown that this task is achievable. A detailed study (**AutomatingLinguisticsBasedCues**
makes a structured approach using linguistic-based cues and draws attention to further studies for automating deception detection. In this study, the authors extend linguistic-based cues with complexity, expressivity, informality, and content diversity. Instead of using humans as cue identifiers, authors use *Natural Language Processing* (NLP) techniques, namely an NLP tool called iSkim (**iSkim_Zhou**), to extract cues automatically. Another study also focuses on linguistic cue analysis. With a small dataset and employing the C4.5 (**C45_Salzberg**) algorithm, the authors reach 60.72% accuracy using 15-fold cross-validation.
Similarly, in (**VerificatoinAndImplementationofLBDeceptionIndicators_Bachenko**), the authors developed a system for automatically identifying 275 truthful or deceitful statements with the use of verbal cues using Classification and Regression Tree (CART) (**ClassificationRegressioniTrees_Breiman**). Additionally, the studies (**OnLyingAndBeingLiedTo_Hancock**; **OnDeceptionAndDeceptionDetection_**
make use of a relatively small dataset and analyze linguistic-based cues. Rubin's series of studies (**OnDeceptionAndDeceptionDetection_Rubin**; **IdentificationOfTruth_R**
TruthAndDeception_Rubin; **TowardsNewsVerification_Rubin**) makes use of Rhetorical Structure Theory (RST) and Vector Space Modeling (VSM). The first captures the coherence of a story using functional relations among meaningful text units and delivers a hierarchical structure for each news story (**RST_William**). The second is the way to represent rhetorical relations in high-dimensional space. The authors utilized logistic regression as their classifier and reached 63% accuracy.
Furthermore, a study from Afroz and colleagues (**DetectingHoaxesFraudsAndDeception_Afro**
investigates stylistic deception and uses lexical, syntactic, and content-specific features. Lexical features include both character- and word-based features. Syntactic features represent sentence-level style and include fre-

quency of function words from LIWC (**LIWC2007_Pennebaker**), punctuation, and POS tagging in which a text is assigned its morphosyntactic category (**POS_Daelemans**). Finally, content-specific features are keywords for a specific topic. For classification, the authors then leveraged Support Vector Machines (SVM) (**SVM_Hearst**). More comprehensive and modern approaches such as (**LiarLiarPantsOnFire_Wang**) also leveraged the power of *Convolutional Neural Networks* (CNNs) to determine the veracity of news.

- *Objectivity-oriented*: Objectivity-oriented news content models aim to detect indicators of the lessening of objectivity in news content (**FakeNewsDetectionOnSocialMediaA** These indicators are observed in the news from misleading sources, such as hyperpartisan sources which display highly polarized opinions in favor of or against a particular political party. Consequently, this polarized behavior motivates the fabrication of news that supports the sources' political views or undermines the opposing political party. *Hyperpartisan news* are a subtle form of fake news and defined as misleading coverage of events that did actually occur with a strong partisan bias (**FightingMisinformationOnSocialMedia_Pennycook**). Since the spread of hyperpartisan news can be detrimental, many approaches to detect hyperpartisanship in news articles have been developed. For instance, in (**AStylometricInquiry_Potthast**), the authors take a stylometric methodology to detect hyperpartisan news. In this study, the authors employ 10 readability scores, and dictionary features where each feature represent the frequency of words from a carefully crafted dictionary in a given document with the help of General Inquirer Dictionaries (**TheGeneralInquirer_Stone**). A competition for detecting hyperpartisan news (**SemEvalHyperpartisanNewsDetection_Kiesel**) hosted several teams with a variety of ideas which include the utilization of n-grams, word embeddings, stylometry, sentiment analysis etc. The most popular method was the usage of embeddings, particularly the models that leveraged BERT (**BERT_Devlin**).

  Also used for dissemination of hyperpartisan news (**SemEvalHyperpartisanNewsDetection_Ki** another form of fake news that is evaluated under this focus is *Yellow-journalism*, which utilizes clickbaits such as catchy headlines, images etc. that invokes strong emotions, and it aims to generate revenue (**ClickbaitDetectionUsingDL_Agrawa** **ClickbaitAndTabloidStrategies_Dolors**). Studies that aim to detect clickbaits mainly focus on headlines. For example, in (**DivingDeepIntoClickbaits_Rony**), the authors construct a DNN in which they use distributed subword embeddings (**EnrichingWordVectorsWithSubwordInfo_Bojanowski**; **BagOfTricksForTextClassificat** as features with an extension of skip-gram model (**DistributedRepresentationsOfWords_Mikol**

Knowledge-based: Being the most direct way of detecting fake news, these approaches make use of external fact-checkers to verify the claims in news content (**FakeNewsDetectionOnSocia** Fact-checkers are either sophisticated algorithms, domain experts or crowdsourced to assess the truthfulness of a claim in a specific context (**FactChecking_Vlachos**). With growing attention on fake news detection, automated fact-checking has drawn much attention, and considerable efforts have been made in this area (**AutomatedFactCheckin** **OverviewOfCheckThat_Barroncede**). We categorize knowledge-based news content models as *Expert-oriented*, *Crowdsourcing-oriented*, and *Computational-oriented*.

- *Expert-oriented*: These approaches are essentially dependent on human domain experts who investigate the integrity of a news piece collecting relevant information and documents to come up with a decision about the truthfulness of a claim [2]. Platforms like Politifact [3] and EUfactcheck [4] are examples for expert-oriented fact-checking for all news from a variety of sources. These platforms label news in a range such that the label reflects the veracity of the news. A different approach for labeling is exercised by Snopes [5], which extends the same logic of Politifact by including different aspects of fact-checking such as Scam, Miscaptioned, Outdated, etc [6]. Recently replaced by an irrelevant magazine website, another instance was Gossipcop [7], which dealt with celebrity fact-checking and contributed to the creation of fake news dataset (**FakeNewsNet_Shu**). Even though expert-based fact-checking is reliable, with the increasing magnitude of news stream and speed of spread, it is not scalable to fact-check every news piece by hand; thus, manual validation alone becomes insufficient (**ASurveyOnAutomatedFactChecking_Guo**).
- *Crowdsourcing-oriented*: Powered by the wisdom of crowds (**WisdomOfCrowds_Galton**), crowdsourcing-oriented fact-checking is a collection of annotations that are afterward aggregated to obtain an overall result indicating the veracity of the news. Unlike professional fact-checkers, who are in short supply, this approach is scalable given that the crowd contains enough literate people (**ScalingUpFactChecking_Allen**). For instance, Twitter launched a program called Birdwatch [8], in which the users are able to leave notes for tweets that they think contain misinformation. Furthermore, this tool allows

[2] https://www.politifact.com/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/

[3] https://www.politifact.com/

[4] https://eufactcheck.eu/

[5] https://www.snopes.com/

[6] https://www.snopes.com/fact-check-ratings/

[7] https://web.archive.org/web/20190807002653/https://www.gossipcop.com/about/

[8] https://twitter.github.io/birdwatch/overview/

users to rate each other's notes, leading to the diversity of perspectives [9]. Another example is from Facebook [10], which uses a third party of crowdsourced fact-checkers called International Fact-Checking Network [11] (IFCN).

- *Computational-oriented*: Heavily dependent on external sources, computational-oriented models are scalable, automated systems that are designed to predict whether a claim is truthful or not. The studies that focused on this type of approach mainly try to solve two issues: (i) identifying check-worthy claims and (ii) estimating the integrity of claims (**FakeNewsDetectionOnSocialMediaADataMiningPerspe** The first issue requires the extraction of factual claims from news content or other related textual content. For example, in (**DetectingCheckWorthyClaims_Hassan**), the authors collect presidential debate transcripts, then label them into three classes with the help of crowdsourcing. Using annotated data and supervised learning techniques, the authors uncover some interesting patterns in these transcripts. Another study that covers both issues uses Wikipedia information to generate factual claims and then check whether a given claim is truthful or not (**FEVER_Thorne**). The second issue, compared to the first one, requires the utilization of structured external sources. *Open web* and structured *knowledge graphs* are the two most prominent tools when tackling this issue. Open web tools analyze features like mutual information statistics (**UnsupervisedNamedEntityExtraction_Etzioni**), frequency, and web-based statistics (**WebBasedStatisticalFactChecking_Magdy**). On the other hand, knowledge graphs are interconnected. One noteworthy example is ontologies such as DBPedia (**DBPedia_Auer**), using which one can define semantic relations and rules in order to infer whether a claim is correct (**SemanticFakeNewsDetection_Bracsoveanu**).

Social Context Models. The interconnected design of social media can be leveraged by extracting user-based, post-based, and network-based features and utilizing these features to supplement news content models. Social context models exploit related user engagements for a news piece by capturing this external information from multiple angles. Two types of social context models are prominent: *Stance-based* and *Propagation-based* (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**).

- *Stance-based*: Given a news piece, these approaches estimate the user's stance toward a specific news topic. More formally, stance detection in social media deals with users' viewpoints toward particular topics by means of various aspects related to users' posts and characteristic traits (**StanceDetectionOnSocialMeda_Abeer**).

[9] https://twitter.github.io/birdwatch/diversity-of-perspectives/

[10] https://www.facebook.com/formedia/blog/third-party-fact-checking-how-it-works

[11] https://www.poynter.org/ifcn/

The user's stance information can be extracted either implicitly or explicitly. Implicit stance can be automatically obtained from social media posts with the help of NLP tools such as sentiment analysis (**StanceAndSentimentINTweets_Saif**). Explicit stances are rather easier to obtain since they are direct expressions of opinions or emotions. For example, "like" on Twitter or Facebook, "upvote" and "downvote" ratings on Reddit, and "like" and "dislike" on Youtube are explicit stances of users. A study that utilizes explicit stances called Some Like it Hoax, uses logistic regression and harmonic boolean label crowdsourcing for classification on a dataset they curated from Facebook. In the stance classification process, they consider the likes and the issuer of likes for each post. They state that logistic regression comes short in this task since it cannot learn anything about posts without likes (**SomeLikeItHoaxDataset_Tacchini**). An early example of implicit stance detection leverages the dialogic relations between authors by constructing graphs that represent the interaction between authors. They show that this information can improve the performance of stance-detection models (**StanceClassificationDialogicProps_Walker**). A more detailed study (**StanceClassificationOnTwitterDebates_Addawood**) investigates stance classification considering lexical, syntactic, twitter-specific and argumentation feature types. Although some twitter-specific features can be considered as explicit stances, such as if the tweet is a retweet, if a tweet contains the title to an article, if a tweet contains a hashtag, etc., those features are later aggregated before it is fed to the classifier. The authors reach the highest F1 score using lexical and argumentation features. In literature, there are also implicit stance-based approaches that aim to detect the veracity of a news piece by exploring the relationship between a headline and the article (**ARetrospectiveAnalysisOfFNC_Hanselowski**; **StanceDetectionInFakeNews_Ghanem**).
Another variation of stance-based detection is rumor detection. One example of a rumor detection model is a Bayes classifier that utilizes content-based, network-based, and twitter-specific meme features through *Information Retrieval* (IR) techniques (**RumorHasIt_Qazvinian**). In this study, the authors propose a general framework that leverages statistical models and maximizes a linear function of log-likelihood ratios to retrieve rumorous tweets. They show that the features they used contribute to their model's overall performance.

- *Propagation-based*: Inspired by the assumption that the veracity of a news event is highly correlated with the credibilities of related social media posts, propagation-based models employ the interrelations of related social media posts to classify a news piece as truthful or not (**FakeNewsDetectionOnSocialMediaADataMiningPerspective_Shu**). These models can be based on either *homogeneous networks* or *heterogeneous net-*

works. Homogeneous networks are built upon a single type of entity, such as a post or event (**NewsVerificationByExploitingConflictingSocialViewpoints_Jin**). A study by **NewsVerificationByExploitingConflictingSocialViewpoints_Jin** created homogeneous credibility networks for each topic which is extracted using an unbalanced version of the Joint Topic Viewpoint Model (**FindingAndArguingExpressions_Trabelsi**). These credibility networks consist of nodes as tweets and edges as links, defined by either supporting or opposing. On the other hand, heterogeneous networks can contain multiple types of entities such as events, sub-events, posts, comments, etc. For example, in (**NewsCredibilityEvaluationOnMicroblog_Jin**), the authors build a hierarchical propagation graph that contains events, sub-events, and messages from parent to child, respectively. Using an iterative method, they provide a globally optimal solution for the graph optimization problem in the study. Furthermore, an interesting study from a decade ago based its credibility estimation algorithm on PageRank and similarity scores. Their propagation network consists of graphs in which possible nodes are events, tweets that were posted about that event, and users who posted those tweets. A more recent study, which we also use in this thesis, is *User Preference-Aware Fake News Dataset* (UPFD) (**UPFD_Dataset_Shu**). This dataset houses two different datasets, one from Politifact and one from Gossipcop. Its root nodes are news pieces, the child of the root node are the users who retweeted the news piece, and the children of the child node are the users who are assumed to have retweeted the news piece after its parent in terms of time. The authors use news content and social engagement information to construct the graph. The best-performing model is based on news and social context. It uses GraphSAGE (**GraphSAGE_Hamilton**) as graph encoder and BERT (**BERT_Devlin**) as the text encoder and reaches 84.62% and 97.23% accuracy on Politifact and Gossipcop, respectively.

We examined two types of FND models, namely, news content and social context models. For each type, we further categorized then defined each type of model, and we gave examples for each. It is crucial to note that approaches are not necessarily purely news content or social context-based; they can be based on both news content and social context. For instance, like the example we gave in propagation-based social context models, GraphSAGE, there are models such as GNN-CL (**GraphNeuralNetworksWithContinualLearningFakeNe** or GCNFN (**FakeNewsDetectionUsingGeometricDeepLearning_Monti**), which are baseline models for UPFD (**UPFD_Dataset_Shu**) and will be discussed in detail in the next section.

To summarize this section, we have introduced the history and definitions of fake news in subsection **??**. Then, we investigated the foundations of fake news and gave motivations for developing automated FND systems in subsection **??**. Following that,

we examined the evolution and characterized FND models in section **??**. We have included at least two examples for each type of model and briefly summarized their approaches. We also briefly examined one of the datasets (**UPFD_Dataset_Shu**) and models (**GraphSAGE_Hamilton**) used in this thesis; however, in-depth information will be provided in the next chapter.
In **??**, we elaborate on the techniques available in explainable artificial intelligence. We discuss the qualities of a reasonable explanation, and we highlight the importance of the interpretability of a model. We give essential definitions that will be used throughout this thesis.

2.2 Explainable Artificial Intelligence

Understanding and interpreting a model's prediction is very important nowadays since this understanding allows to validate the reasoning of the model and extract rich information for a human expert, and can lead to increased trust in the model (**WhyShouldITrustYou_Riberio**). Furthermore, explanation of a model can help to improve the model (**AUnifiedApproach_Lundberg**) and alleviate concerns raised by Ethical AI (**MachineBias_Angwin**; **EURegulationsOnDecisionMaking_C**
In this section, we introduce the background for XAI techniques that were used in this thesis. First, in **??** we characterize XAI by following works from **TheMythosOfModelInterpretability_Lipt**
and **XAIConceptsTaxonomies_Arrieta** and give definitions to clarify the taxonomy. Then, in **??**, we discuss the properties of good explanations, the goals of XAI and the evaluation techniques for explanation ethods. Finally, in **??** we briefly lay out the most frequently mentioned explanation methods in the literature, along with the ones we use in this thesis. We summarize each of them and cover explanation techniques offered to any kind of neural network.

2.2.1 Foundations of Explainable Artificial Intelligence

Initial AI methods were not sophisticated enough to require additional explanation schemes. In the last years, expanding applications of DNNs have led to the adoption of these opaque systems even more. Although empirically successful thanks to enormous parameter spaces and numerous layers, DNNs are complex *black-box* models in terms of interpretability (**CanWeOpenTheBlackBoxOfAI_Castelvecchi**).
In the XAI context, *black-box* or *opaque* models are considered to be the opposite of *transparent* because they require a further search to understand their inner workings (**TheMythosOfModelInterpretability_Lipton**). Accordingly, humans hesitate to use systems that are not directly interpretable and reliable (**xAIForDesigners_Zhu**), making *interpretability* essential. Moreover, from a legal perspective, the notion *right to explanation* brings more attention to interpretabilty (**TheMythosOfModelInterpretability_Lipton**).

Particularly in situations such as when:

- The prediction of AI directly affects human life, e.g., fully autonomous cars in traffic, medical AI assistants etc.

- The reasons behind an AI system's decision can not be clearly determined.

With the additional demand from the Ethical AI field (**EURegulationsOnDecisionMaking_Goodman**), the research community has put in a great amount of effort to gap the bridge between a black-box model and its interpretability. However, the lack of consensus on taxonomy has led to synonymous usages of interpretability. The early definitions for interpretability were too broad, describing it as essentially an additional design driver when building a model (**TheBayesianCaseModel_Kim**) or a requirement for *trust* (**InteractiveAndInterpretableMLModels_Kim**). But can trust be defined in an objective way? Is the accuracy or F1 score of the model is enough to trust a model? To answer the first question, **TheMythosOfModelInterpretability_Lipton** argues that trust is subjective and it is not technically defined. To answer the second question, taking only the performance metrics as a baseline for trust in the model is shown to be an incorrect approach, particularly studies that analyze models with *adversarial examples* (**DetectingAdversarilaImageExamples_Bin**; **AdversarialExamples_Yuan**). Moreover, **TowardsARigorousScienceML_Velez** argues that the need for interpretability comes from the *incompleteness* of the problem formalization.
Instead of trying to find a technical definition for interpretability, we can categorize existing systems in terms of their transparency. **TheMythosOfModelInterpretability_Lipton** states two properties for interpretable models: *transparency* and *post-hoc interpretability*. The definition for the first and its related terms are given as in the following,

Definition 2.2.1 (*Understandability*)**.** *"Denotes the characteristic of a model to make a human understand its function - how the model works - without any need for explaining its internal structure or the algorithmic means by which the model processes data internally"* (**MethodsForInterpretingAndUnderstandingDNNs_Montavon**; **XAIConceptsTaxonomies_Arriet**

Definition 2.2.2 (*Transparency*)**.** *"A model is considered to be transparent if by itself it is understandable"* (**XAIConceptsTaxonomies_Arrieta**).

To elaborate further, we discuss degrees of transparent models as not all models provide the same extent of understandability (**XAIConceptsTaxonomies_Arrieta**). Both in **TheMythosOfModelInterpretability_Lipton** and **XAIConceptsTaxonomies_Arrieta** the categorization is made as: *simulability*, *decomposability*, and *algorithmic transparency*. We discuss each of them briefly.

- *Simulability*: Denotes the model's characteristic to be simulated or thought only by a human. Thus, the complexity of a model plays an important role here. Models that can be presented to a human in terms of text and visualizations are considered interpretable (**WhyShouldITrustYou_Riberio**), and in this case, models this elementary fall into simulatable models category (**XAIConceptsTaxonomies_Arrieta**; **RegressionShrinkage_Tibshirani**).

- *Decomposability*: Represents the model's characteristic to explain each part of the model. Basically, when all components of a model are simulable, then that model is decomposable (**TheMythosOfModelInterpretability_Lipton**) given that the inputs are already interpretable (**XAIConceptsTaxonomies_Arrieta**).

- *Algorithmic Transparency*: Deals with the user's comprehension of the input's journey from entering the model to becoming a prediction (**TheMythosOfModelInterpretability_Lipton** **XAIConceptsTaxonomies_Arrieta**). For example, linear models can be considered algorihtmically transparent since the user can understand how the model can act in a given situation (**AnIntroductionToStatisticalLearning_Gareth**).

The second property of interpretable models, *post-hoc explainability*, aims to improve the interpretability of not readily interpretable models. It does so by means of *text explanations*, *visual explanations*, *local explanations*, *explanations by example*, *explanations by simplification*, and *feature relevance explanations* techniques (**TheMythosOfModelInterpretability_Lipton**; **XAIConceptsTaxonomies_Arrieta**). In a more general sense, post-hoc explainability methods can be grouped into three categories in terms of the knowledge of the target model, granularity of focus, and form: *model-specific or model-agnostic*, *local or global* and *form*. The first category refers to the explainability method's assumption on the model's structure. *Model-specific* techniques can be utilized with a limited set of models since these techniques make an assumption on the model to be explained. On the other hand, *model-agnostic* techniques are designed in such a way that does not require knowledge about model's inner workings (**XAIConceptsTaxonomies_Arrieta**; **ASurveyOfMethodsForExplainingBlackBoxModels_Guidotti**). The second category denotes the explanation's domain. *Local* explanations reason about a particular prediction of a model at feature level (**TowardsARigorousScienceML_Velez**) (e.g. compute a saliency map by taking the gradient of the output with respect to a given input vector (**TheMythosOfModelInterpretability_Lipton**)), whereas global explanations aim to outline the model's general behavior on the dataset (**XAIConceptsTaxonomies_Arrieta**; **ASurveyOfMethodsForExplainingBlackBoxModels_Guidotti**; **TowardsARigorousScienceML_Velez**). Global explanations are usually presented in the structure of a series of rules (**InterpretableDecisionSets_L** The third and last category, form of the explanation is the manner that is conveyed to the user. We will discuss specific forms of explanation in detail after we give a definition of *explainability*, since from now on we talk about the explainability of a model rather than its interpretability.

Definition 2.2.3 (Explainability)**.** *"Explainability is associated with the notion of explanation as an interface between humans and a decision maker that is, at the same time, both an accurate proxy of the decision maker and comprehensible to humans."* (**XAIConceptsTaxonomies_Arrieta**; **ASurveyOfMethodsForExplainingBlackBoxModels_Guidotti**)

- *Text explanations* are techniques that learn to produce textual expressions that assist user to understand the outcomes of the model (**TowardsExplainableNeuralSymbolic_Bennetot**).
- *Visual explanations* are techniques that supplement a model's explainability by visualizing the model's behavior. Due to the mismatch between high-dimensional nature of complex ML systems and the capacity of human reasoning (**HowTheMachineThinks_Burrell** visual explanations often employ dimensionality reduction practices (**XAIConceptsTaxonomies_Arri**

From the perspective of explainability, one would intuitively prefer transparency since transparent models can be explained with ease. However, some works argue that as transparency of a model increases their performance usually tends to decrease (**ExplaniableAIASurvey_Dosilovic**), although other works argue that this is not necessarily true, particularly in cases where the data is well structured and the quailty and value of available features is outstanding (**StopExplainingBlackBoxmodels_Rudin**). Furthermore, considering FND models, the need for big and complex models can not be avoided since news pieces tend to be long texts, or social networks are represented as graphs, thus forcing SOTA FND models to utilize complex approaches that decreases transparecy such as word embeddings, data fusion, graph data structure (**UPFD_Dataset_Shu**).
On the other hand, although global explanations can be helpful to domain experts by providing information about what model has learnt on a global level, it might be difficult to obtain (**TheMythosOfModelInterpretability_Lipton**). Instead, local explanation methods are more easier to obtain and more practical for real-world applications. For example, if a user requests an explanation for a prediction, local explanations can provide it, which also complies with "right to explanation" (**EURegulationsOnDecisionMaking_Goodman**).
Depending on the model adopted for FND, we might be required to use model-specific or model-agnostic approaches. For instance, when dealing with a GNN, model-agnostic approaches do not provide easily interpretable explanations, thus requiring a model-specific explanation method. On the other hand, when dealing with a DNN model-agnostic post-hoc approaches are usually the choice (**XAIConceptsTaxonomies_Arrieta**).
Therefore, for FND models used in this thesis, we are required to adopt both model-agnostic and model-specific post-hoc approaches. Below we list types of post-hoc approach as listen in (**XAIConceptsTaxonomies_Arrieta**).

- *Explanations by example*, a method suggested by **CaseBasedExplanation_Caruana**, focus on obtaining representative information from a model by providing explanations for an example that sufficiently illustrates the inner workings of the model (**HowToExplainIndividualClassificationDecisions_Baehrens**; **XAIConceptsTaxonomies_Ar**
- *Explanations by simplification* techniques construct a new and simplified system to provide explanations for a model. These simplified systems keep the performance of the original model while displaying less complexity (**XAIConceptsTaxonomies_Arrieta**).
- *Feature relevance explanations* compute feature relevance scores for a model's variables in order to determine the effect of a feature has upon a model (**XAIConceptsTaxonomies_Arriet**

We discussed what kind of explanation methods we can adopt and how these methods can shape the design of a model and the forms of explanations that aims to convey information about the model's behavior. It is possible to see a combination of the previously mentioned explanation forms. In order to present the user with an comprehensible explanation, we characterize a good explanation and define its important properties in the next subsection.

2.2.2 What Makes A Good Explanation

In literature, the requirements for a *good explanation* are rigorously researched. However, there is not a clear definition of how an explanation should look like or convey to the user. It is challenging to objectively define what makes a good explanation (**XAIConceptsTaxonomies_Arrieta**). In order to tackle the issue of subjectivity of explanations, XAI draws wisdom from social and cognitive sciences. A comprehensive study on social sciences and XAI by **ExplanationInAI_Miller**, analyzes explanations in terms of the content, the explainer and the explainee. The author argues that the research in AI is lacking knowledge about the properties and structure of an explanation. The major findings on how a good explanation should be are outlined below.

- Explanations are *contrastive* (**ContrastiveExplanation_Lipton**; **XAI_BewareInmatesRunningTheAsy** When presented with counterfactual explanations, users can understand the decision made by the model easier (**ExplainableAndInterpretableModels_Escalante**; **MLCVPatternRecognition_Lopez**; **MLCVPatternRecognition_Lopez**; **CounterfactualsInXAI_Byr** For example, rather than asking why event A occurred, we ask why event A occurred instead of an event B (**ExplanationInAI_Miller**; **XAIConceptsTaxonomies_Arrieta**).
- Explanations are *selective*. Presenting all the causes for an event to a human is pointless, since humans are inclined to select a couple main causes out of numerous, sometimes countless, causes as shown in (**ExplainingCollaborativeFiltering_Herlocker**).

Accordingly, **ExplanationInAI_Miller** argues that this selection process is shaped by specific cognitive biases.

- Explanations are *social*. They are conveyed from explainer to explainee via a social interaction. Hence, explanations are transferred through the frame of explainer's beliefs about the explainee's beliefs (**ExplanationInAI_Miller**).
- Probabilities probably don't matter. Even though probabilities matter when creating the explanations, the usage of these statistical relations in explanations is not as effective as that of causes. If the underlying causal explanation is not included, then the utilization of statistical generalisations is not sufficient (**ExplanationInAI_Miller**).

It should be noted that the characteristics of a good explanation is not limited to the ones mentioned above. These are the most prominent characteristics of numerous which is discussed in (**ExplanationInAI_Miller**) detail. An important aspect here is that the explanations are provided to an *explainee* who is the *audience* in (**XAIConceptsTaxonomies_Arrieta**), which refers to the person receving the explanation. It is further noted in (**XAIConceptsTaxonomies_Arrieta**) that explanations are dependent on the audience, i.e., an explanation meant for an end-user will not be enough for a domain expert or an explanation for a domain expert might be too complicated for an end-user. Also, we will refer to the expainee as *audience* from now on.

Main target audience is the main driver when considering the needed outcomes for an explanation. **XAIConceptsTaxonomies_Arrieta** summarizes the pursued goals when trying to attain explainability. These are listed below.

- *Trustworthiness* deals with the assurance of a model's intended behavior when the model is presented with real-world scenarios (**TheMythosOfModelInterpretability_Lipton**; **StructuringDimensionsForCollaborative_Antunes**). Some studies highlight the importance of *trustworthiness* as a requirement for explainability (**WhyShouldITrustYou_Riberio**; **InteractiveBayesianCaseModel_Kim**). The main target audience for this goal is domain experts, and users affected by the model decisions (**XAIConceptsTaxonomies_Arrieta**).
- *Confidence* refers to the robustness and stability of a model (**XAIConceptsTaxonomies_Arrieta**). **Stabi** argues that stability is a prerequisite when obtaining explanations from a model. Moreover, (**XAIConceptsTaxonomies_Arrieta**) argues that trustworthy explanations should not be obtained from unstable models. The audience relevant for this goal are domain expers, developers, managers, and regulatory entities.
- *Fairness* refers to a model's potential to ensure a fair prediction for a user affected by the model's prediction on the basis of characteristics such as age, race, gender,

etc (**XAIConceptsTaxonomies_Arrieta**; **FairnessInML_Oneto**). The audience for this goal consists of users affected by model decisions and regulatory entities.

- *Transferability* refers to the model's capability to perform on unseen data. It is a desired goal to have not just for explainability but also for obtaining a good performance from the model (**AppliedPredictiveModeling_Kuhn**). The audience for this goal is domain experts and data scientists.
- *Causality* denotes the causal relationships between variables of a model (**Causality_Pearl**). It aims to provide causal information among the data variables. Its main audience is domain experts, managers, and regulatory entities (**XAIConceptsTaxonomies_Arrieta**).
- *Informativeness* is a goal meant for all users and it deals with the information provided by the model. In order to fill the gap between user's decision and the prediction of a model, a massive amount of information about the problem at hand needs to be conveyed to the end user (**XAIConceptsTaxonomies_Arrieta**).
- *Accessibility* refers to the possibility of end users getting more involved in a model's development or improvement (**XAI_BewareInmatesRunningTheAsylum_Miller**). The audience for this goal includes product owners, managers, and user affected by model decisions.
- *Interactivity* denotes a model's capability to interact with the user (**InteractiveAndInterpretableMLM** The main audience consists of domain experts and users affected model decisions.
- *Privacy awareness* is a goal not frequently seen in the literature. It deals with the learnings of a model's internal representation such that these learnings might pose a privacy breach. From the opposite perspective, it is a differential privacy breach when an unauthorized third party can explain the inner workings if a trained model (**XAIConceptsTaxonomies_Arrieta**).

On the other hand, for a given explanation and its preferred characteristics, how do we objectively evaluate an explanation? For example, considering a model, the evaluation metrics obtained from the test set reflect the model's overall performance on unseen data and allow to compare different models that use the same dataset (**PMLB_Olson**). For example, metrics like accuracy, F1 score, recall, and precision are often used in the evaluation of models. Given that there are numerous metrics, it should be noted that different domains and models may require different evaluation metrics (**BeingAccurateIsNotEnough_McNee**; **AReviewOnEvaluationMetrics_Hossin**; **PeeringIntoTheBlac** For a comprehensive study on the evaluation metrics of ML models, please refer to (**EvaluatingLearningAlgorithms_Japkowicz**). Similar to models, explanations require evaluation methods that can quantify their performance. So far, we have seen that explanations might have different audiences, they can take several forms, and they have desired properties. Therefore, like models, there should be a set of explanation evaluation methods which focus on different categories of explanation approaches. In fact, a rigorous study by **TowardsARigorousScienceML_Velez** lays out the categorization of explanation evalution approaches. The authors split the evaluation methodologies into three:

1. *Application-grounded evaluations* involve conducting experiments on real humans who are domain experts interacting with explanations in a real-world application. This kind of evaluation directly tests the objective of the system, thus, attaining high performance with respect to application-grounded evaluation suggests good evidence of the explanation's success. The fact that we need humans interacting with a real-world application in an environment which can be observed for experimentation makes this type of evaluation more specific and thus the most costly of all three types of evaluation (**TowardsARigorousScienceML_Velez**).

2. *Human-grounded evaluations* are constructed by simpler experiments conducted on real humans who are not necessarily domain experts. Although this type of evaluation is less specific compared to the application-grounded evaluations, it offers more flexibilty and less costly. It is a good choice when the task is to test the quality of an explanation in a general sense (**TowardsARigorousScienceML_Velez**). For example, a recent study (**AHumanGroundedEvaluationBenchmark_Mohseni**) used human attention maps that overlay on images as explanations and asked users to rate the decision made by the model. The study further argues that the evaluation on these attention maps can be utilized to understand the *trustworthiness* of a model.

3. *Functional-grounded evaluations* do not include real humans, instead this kind of evaluations use a formal definition of interpretability as a proxy to assess the explanation's quailty. The lack of human dependency makes them favorable due to the low cost. Typically, these evaluations are preferred when conducting experiments with humans in the loop might be unethical. The challenge with these evaluations is to select the right proxy models. Accordingly, when possible, it is considered good practice to first obtain proxies that were verified, for instance, by human-grounded evaluations (**TowardsARigorousScienceML_Velez**).

From high cost to low cost, and more specific to more broad, one can opt for an evaluation technique to obtain a performance indicator of an explanation. As discussed above, each approach require a completely different setting, which brings their shortcomings with it. For instance, depending of the availability of resources such as time, finances, expertise of the user or sufficiency of human subjects one might have to opt for a

different evaluation technique.
Having highlighted important characteristics of a good explanation, we now move forward to the frequently mentioned techniques used in XAI. We mostly focus on post-hoc local explanation techniques and outline their contribution to this thesis.

2.2.3 Overview of Techniques in Explainable Artificial Intelligence

As discussed in the last section, when contructing an explanation method, one has to consider the audience, opt between model-specific or model-agnostic, local or global explanations, and also, utilize various forms of explanation. In literature, there exist various combinations of previously mentioned options. For transparent models, no further explanation method needed, one can obtain relevant information in the forms of weights or attention scores, given that the features are simple enough (**XAIConceptsTaxonomies_Arrieta**). In particular, we talk about explanation methods that were frequently mentioned in studies and relevant for this thesis.
First, we discuss the initial methodologies aimed to gain insight from a black-box model. The one of the initial approaches was to ask the question: *What happens if we remove this part of the input? Sensitivity Analysis* (SA) deals with analytical assessment of the effect of an omitted input variable on the uncertainty of a model (**SensitivityAndGeneralizationInNNs_Novak**). SA can be done on two levels, local and global. *Local Sensivity Analysis* (LSA) assesses the impact of the changes in the input on the output whereas *Global Sensitivity Analysis* (GSA) examines the effect of each variable (feature) with respect to the variations of all parameters (**InputPerturbationSensitivity_Rao**). In literature, there are a variety of approaches for both GSA and LSA. For instance, **SensitivityAndGeneralizationInNNs_Novak** constructs a GSA that employs the partial derivative of each parameter in the back-propagation algorithm to explore the change rule, which admits the *Input-Perturbation-Sensitivity* (IPS) that allows to obtain global sensitivity. An interesting example of a GSA and LSA fusion approach, **SensitivityAnalysisForPNNs_Kowalski**, utilizes LSA to reduce the number of input features and GSA to reduce the number of patterns learned by a model.
Another approach was to calculate relevance scores for each feature using saliency maps. The usage of saliency maps first appeared in CNNs for images (**DeepInsideCNNs_Simonyan**), then extended to NLP in **AskTheGRU_Trapit**; **ExtractionOfSalientSentences_Denil**. Typically, salience maps compute a gradient to get a relevance score to an input feature. In other words, they convey information about the model's sensitivity with respect to the input.
A popular method used in XAI is *Layer-wise Relevance Rropagation* (LRP) which was first introduced for *Fully Connected Networks* (FCNs) and CNNs in **LRP_Lapuschkin**, then extended to *Recurrent Neural Networks* (RNNs) in **ExplainingRNNs_Arras**. LRP

assumes that a model can be *decomposed* into several layers which can contain feature relevant information. From the last layer to the input layer, LRP computes a relevance score for each dimension of the vector at a layer, and as LRP moves backwards in the layers, the sum of relevance scores do not change, staying always equal to the prediction probability (**LRP_Lapuschkin**).

Similar to LRP, a study for explaining DNNs offers another solution named *Deep Learning Important FeaTures* (DeepLIFT) (**DeepLIFT_Shrikumar**). This approach addresses two shortcomings of LRP, namely, the failure to model saturation caused by activation functions, and the possibility of getting misleading importance scores due to discontinuous gradients. Combining techniques from LRP and integrated gradients (**GradientsOfCounterfactuals_Sundararajan**), DeepLIFT computes importance scores based on the *difference-from-reference* approach which allows propagation of information even if the gradient is zero. Difference-from-reference is a method which involves determining a reference then getting the difference between the reference and the output. This method is also later adopted by to create DeepSHAP (**AUnifiedApproach_Lundberg**).

In contrast to model-specific approaches like LRP and DeepLIFT, *Locally Interpretable Model-agnostic Explanations* (LIME), as the name suggests, is a model-agnostic method. LIME interprets the predictions of any black-box model by approximating the model around a prediction. This approximation allows to obtain a locally faithful and interpretable version of the model (**WhyShouldITrustYou_Riberio**).

So far, there is no study that unifies all the works to create one explainability framework. To address this lack of unification, **AUnifiedApproach_Lundberg** offers *SHapley Additive exPlanation* (SHAP) framework, in which the authors utilize recent studies from game theory based on (**GameTheory_Shapley**). These studies are *Shapley regression values* (**AnalysisOfRegressionInGameTheory_Lipovetsky**), *Shapley sampling values* (**ExplainingPredictionModels_Strumbelj**)*Quantitative input influence* (**AlgorithmicTransparencyViaQ**
and recent approaches like LIME, DeepLIFT are utilized to create a model-agnostic and model-specific explainers. SHAP values measure the feature importance and obtained via Shapley values of conditional expectation function of a model (**AUnifiedApproach_Lundberg**).

Model-agnostic SHAP values are computed using Shapley sampling values method which uses an approximation of a permutation adaption of of classic Shapley value estimation. For example, *KernelSHAP* employs LIME with linear explanations and Shapley values to find a weighting kernel that enables regression based estimation of SHAP values. On the other hand, the authors propose *LinearSHAP* which can approximate Shapley values using weights for linear models, and *DeepSHAP* which connects DeepLIFT with Shapley values, *Low-order SHAP*, and *Max SHAP* for model-specific explainers. We will discuss SHAP values further in Chapter **??**.

In literature, there is a lack of explanation methods for GNNs. GNNs require graphs as input and an output for either graph or node depending on the focus of the

task (**DeepLearningOnGraphs_Zhang**). Graphs are capable of representing rich relational information between nodes and the node feature information (**DeepLearningOnGraphs_Zhang**; **GNNsAReview_Zhou**). GNNs are powerful tools that are able to learn relational information between nodes as well as node features, making them a perfect candidate for analyzing social media networks (**BeyondSigmoids_Zang**). In our case, we want to understand how a GNN behaves when classifying fake and real news pieces and their propagation networks. A study by **GNNExplainer_Ying** proposes a model-agnostic approach called *GNNExplainer* to explain predictions made by GNNs. *GNNExplainer* takes a trained GNN, input graph(s) and its prediction(s) ad it returns explanations in the form of subgraph(s) of input graph(s) along with the most influential node feaures for the prediction. These subgraphs are constructed by maximizing the mutual information between the subgraph and the input graph with respect to the prediction (**GNNExplainer_Ying**).
Bearing in mind the FND models and explanation methods discussed one can use LIME, DeepLIFT, or SHAP for news content models which are essentially DNNs with textual data as inputs. Especially for understanding which words or word groups are of the most importance, SHAP provides text plots and easily interpretable importance scores. Therefore, when assessing our choice of news content model, we employ SHAP framework, in particular, DeepSHAP. On the other hand, for GNNs the choice is straightforward as there is only one choice. Although, GNNExplainer can be helpful to identify the most important spreaders of a news piece which will be discussed in Chapter **??**.
To conclude this chapter, it should be noted that due to the numerous studies in the literature, we did not cover all explanation methods, but an extensive study can be found in (**InterpretableMachineLearning_Molnar**). Moreover, we were not able to fully cover the psychological and social background of explanations as we did for fake news, however **ExplanationInAI_Miller** provides a rigorous research in that field. In the next chapter, we elaborate on FND models that were used in this thesis. We show how a neural network produces a prediction for a given input. We share our analysis on both textual and graph datasets. After talking about model parameters and hyperparameters, we evaluate our models and talk about the evaluation process. Also, we examine issues like early fake news detection and model aging, particularly for our SOTA FND models.

3 Fake News Detection Models

The automated detection of fake news on social media comes with its characteristic challenges. First, the fact that fake news are constructed to misguide its consumers makes them hard to distinguish by only using news content. Second, when we include social context into the model, the large-scale and noisy nature of social context data represents another issue (**HierarchicalPropagationNetworksForFND_Shu**). Moreover, from a broader perspective, fake news should be detected before it becomes widespread so that the amount of users affected can be minimized.

In this chapter, we examine how these challenges effect the model and dataset's design. We initially take a look at news content models in section **??**. In first section, we lay out the definitions for the materials used. After we give a detailed analysis of the dataset, we talk about the tokenizer and model itself, discuss its performance on the dataset. In section **??** we investigate social context based and hybrid models. Similar to the first section, we give definitons for the used material, then talk about the dataset and model. In this section, we also examine issues such as early fake news detection and model aging.

3.1 News Content Models

The majority of approaches for FND models utilizes news content. Models that base their predictions only on news content focus on the patterns in the text, especially words or word groups that appear frequently in other instances of the same class. As discussed in Section **??**, there exist a variety of approaches available for news content models, however, due to unavailable or outdated datasets, we were unable to work with most news content models.

3.1.1 Notation and Definitions

Here we introduce the notation utilized in this section. Note that these notations will appear in its context, which will provide concrete examples for each symbol defined in Table **??**.

Using this notation we now define some relevant concepts. First, we talk about terms

x^{raw}	Input news article.
y^{raw}	The label of news article.
T	Tokenizer function
ψ	Label mapping function
x^{tok}	Tokenized news article
y	Vectorized class value.
$\lvert x^{tok} \rvert$	The number of tokens in x^{tok}.
X^{raw}	The space of x^{raw}
X^{tok}	The space of x^{tok}.
Y^{raw}	The space of y^{raw}
Y	The space of y
p	Position of a token in the tokenized article x^{tok}.
x^{tok}_p	A token at position p of the tokenized article x^{tok}
V	Vocabulary: A collection of tokens available to the tokenizer.
f	Classifier function, i.e., FND model.
y^*	Prediction of FND model.
x	Numeric vector of x^{tok}
X	Space of x
$f(x)_y$	Prediction score for class y given input x
l	Index of a layer
$l_{embedding}$	Embedding layer
$a^{(l)}_i$	The value of unit i in layer l
$w^{(l)}_{ij}$	Weight between units i in layer l and j in layer $l+1$
σ	Activation function
L	Loss function

Table 3.1: Notation used in this section.

and definitions for *tokenization*, illustrate the mathematical insight in the tokenization process. First, to build upon a concrete foundation, let us consider a news article x^{raw} fed to the tokenizer.

Definition 3.1.1 (*Tokenizer*)**.** A tokenizer $T : X^{raw} \mapsto X^{tok}$ is a function that maps raw textual data to smaller units called tokens.

A token can be a word, character or a subword. Therefore, we define three types of tokenization techniques:

- *Word tokenization* splits the given text into individual words based on a delimeter such as whitespace, commma, etc. This approach creates a vocabulary (V) from the inputs it was trained on. All words do not appear in the vocabulary are replaced with unknown token ([UNK]), and this concept is called being *Out Of Vocabulary*(OOV). Depending on the task, the size of the vocabulary can grow quite large. The solution for exploding vocabulary sizes was introduced in subword tokenization. The commonly used examples for word tokenizers are Word2Vec (**Word2Vec_Mikolov**) and GloVe (**GloVe_Pennington**).
- *Character tokenization* splits the text into single characters. Since the size of available characters is limited and known, the OOV problem is solved by encoding the unknown word by means of its characters. Although looks like a good solution, the length of tokens can be massive for long texts.
- *Subword tokenization* splits the given text into subwords, also called *n-gram characters*. For instance, comparative words like harder is segmented into hard-er, or superlative words like hardest is segmented into hard-est. The most common method for subword tokenization is *Byte Pair Encoding* (BPE). BPE was introduced by **ANewAlgorithmForDataCompression_Gage** but adapted to word segmentation by **NeuralMachineTranslationOfRareWords_Sennrich**. BPE iteratively merges the most frequently appearing character or character sequences. This approach allows for an efficient space usage thus smaller vocabularies (**NeuralMachineTranslationOfRareWords_Sennrich**).

We say that an input is *tokenized* after it is fed to the tokenizer. A tokenized news article $x^{tok} \in X^{tok}$ is a vector of tokens in which the order of the words and characters in $x^{raw} \in X^{raw}$ are kept.

$$T(x^{raw}) = x = [x_1, x_2, \ldots, x_n], \text{ where } n = |x|.$$

We denote an element of x^{tok} at p-th position as x^{tok}_p, and the number of tokens in x^{tok} as $|x^{tok}|$. Thus, we have the range of p, $1 \leq p \leq |x^{tok}|$. The tokens Furthermore, we denote the space of raw label $Y^{raw} = \{"fake", "real"\}$, with $y^{raw} \in Y^{raw}$. We use a label mapping function $\psi : Y^{raw} \mapsto Y$ that maps raw labels to classes, where $Y \in \mathbb{R}^2$ with,

$$\psi(y^{raw}) = y = \begin{cases} 0, & y^{raw} = "fake" \\ 1, & y^{raw} = "real" \end{cases}$$

In order to feed the input to the model, we need numeric data which can be obtained by numerous techniques. One widely used approach is BoW representation which produces features based on the number of occurrences of a word or token. An alternative BoW representation uses the presence/absence of word instead of frequencies. A more sophisticated approach is *Word2Vec*, which encodes words into numeric values by learning word associations. From the perspective of representation of a word, *Word2Vec* can capture different degrees of similarity between words which allows for preservation of semantic and syntactic relationships (**Word2Vec_Mikolov**). It is clear that the transformation of words into numeric vectors is a very crucial stage for FND since we need to maintain as much contextual information as possible. Yet, the SOTA is an even more sophisticated approach called *Transformer* which is utilized by many language models such as BERT. We will discuss Transformer architecture in **??**. For ease of the notation, we refer to this stage as *Embedding Layer* and denote it with $l_{embedding}$.
The input transformation pipeline is illustrated in **??**.

Definition 3.1.2 (*FND Classifier*)**.** An FND classifier $f : X \mapsto Y$ is a function that outputs a predicted scores $f(x)_y$ for each class y for a given input x.

Definition 3.1.3 (*Prediction*)**.** A prediction y^* is the maximum of predicted scores $f(x)_y$ of an FND classifier.

$$y^* = argmax_{y \in Y} f(x)_y$$

Definition 3.1.4 (*FND Neural Network Classifier*)**.** A neural network classifier is a *classifier* f that comprises of layers l with $1 \leq l \leq L$, where L denotes the number of layers. Each layer has a set of units $a_i^{(l)}$ with i denoting the position of the unit in a layer l. We say that between two units $a_i^{(l)}$ belonging to layer l and $a_j^{(l+1)}$ belonging to layer $l+1$ have a weight value $w_{ij}^{(l)}$ that connects them. Along with a non-linear activation function σ, we can define the value of the j-th unit $a_j^{(l+1)}$ in terms of weights and unit values from previous layer for an FCN, with N is the number of units in layer l.

$$a_j^{(l+1)} = \sigma(\textstyle\sum_{i=1}^{N} a_i^{(l)} w_{ij}^{(l)})$$

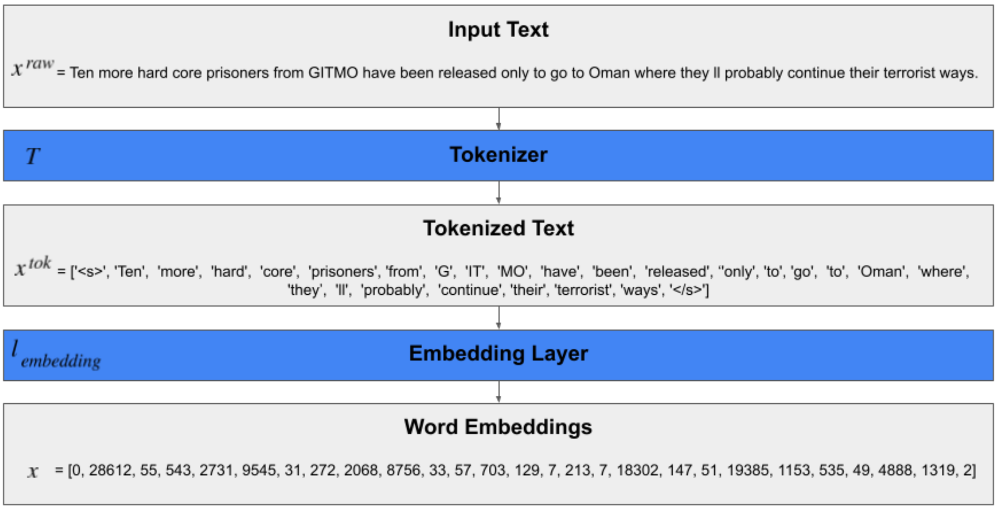
Input Text
x^{raw} = Ten more hard core prisoners from GITMO have been released only to go to Oman where they ll probably continue their terrorist ways.
T
Tokenizer
Tokenized Text
x^{tok} = ['<s>', 'Ten', 'more', 'hard', 'core', 'prisoners', 'from', 'G', 'IT', 'MO', 'have', 'been', 'released', ''only', 'to', 'go', 'to', 'Oman', 'where', 'they', 'll', 'probably', 'continue', 'their', 'terrorist', 'ways', '</s>']
$l_{embedding}$
Embedding Layer
Word Embeddings
x = [0, 28612, 55, 543, 2731, 9545, 31, 272, 2068, 8756, 33, 57, 703, 129, 7, 213, 7, 18302, 147, 51, 19385, 1153, 535, 49, 4888, 1319, 2]

Figure 3.1: The preprocessing pipeline for a textual input.

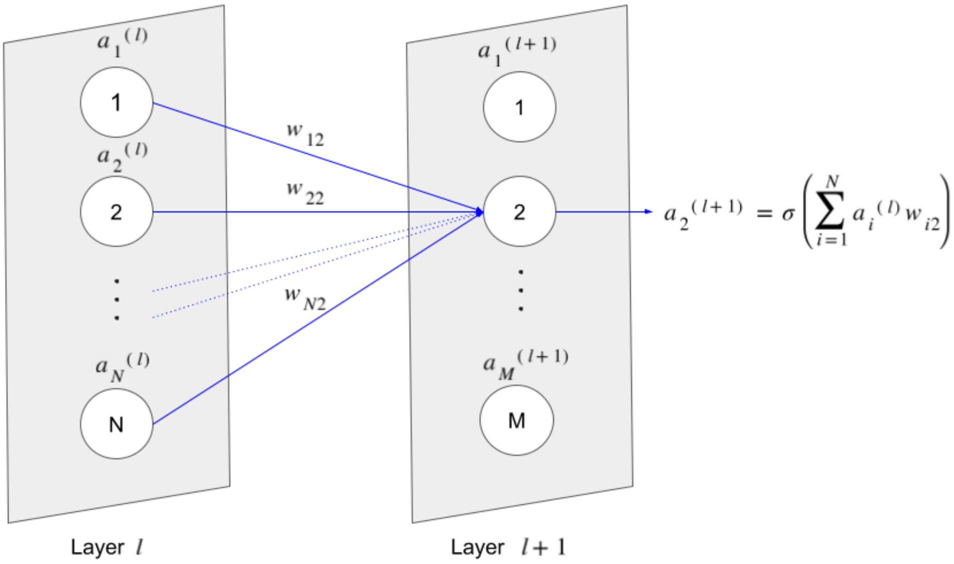

Figure 3.2: Units and layers of an FCN. For brevity, arrows and weights are only drawn for $a_2^{(l+1)}$

Neural networks iteratively optimize the weights between layers such that the produced output is as close as to the expected output. This is done so using optimization methods such as *Gradient Descent* (GD) (**GD_Cauchy**), *Stochastic Gradient Descent* (SGD) (**SGD_Robbins**), *Adaptive Moment Estimation* (Adam) (**Adam_Kingma**) to minimize the loss function L. While there exist many optimization methods available for nueral networks, we do not examine any of them.

For classification problems, we adopt a layer called *Softmax* (**Softmax_Bridle**) outputs the predicted scores $f(x)_y$ for each class y by normalizing outputs $Z_y(x)$ from the previous layer.

$$f(x)_y = \frac{exp(Z_y(x))}{\sum_{\hat{y} \in Y} exp(Z_{\hat{y}(x)})}$$

The FCNs are the simplest architectures for neural networks. In an FCN, all units are connected which means there is a weight between each unit in layer l and $l+1$. While very simple to construct, usage of these architectures when modeling sequences is not preferred due to the number of trained parameters such as weights and biases. Instead, when modeling sequences like sentences and documents, a common approach is RNNs which allow previous outputs to be used while having hidden states. However, RNNs are not powerful enough to represent long-term dependencies (**LearningLongTermDependenciesHard_Bengio**) and suffer from vanishing/exploding gradients (**OnTheDifficultyOfTrainingRNNs_Pascanu**). Utilizing RNNs, an idea was proposed in which these shortcomings were addressed. Called *Long Short-Term Memory* (LSTM) (**LSTM_Hochreiter**), the idea is to keep a cell state which is updated with previous cell's state. This cell state is conveyed to the consequent cells to form a chain that represents the document. More precisely, each cell corresponds to a token whose information will be shared with consequent tokens by the propagation of previously mentioned cell states. This approach is indeed very useful for long documents since news articles tend to be long and their sentences contextually relevant. LSTM is usually used in different variations based on the same idea. For instance, a consequent study has extended LSTMs with *peephole connections* in (**LSTMPeephole_Gers**). LSTMs have been proven to deliver a good performance in NLP tasks such as *speech recognitions* (**AchievingHumanParityinConvSR_Wayne**).

LSTMs perform well however due to long training times and large memory requirements during training, they are being replaced with attention-based models. Having introduced all necessary notation and definitions, in **??**, we discuss the SOTA approach the Transformer models which are based on attention mechanism.

3.1.2 Transformer Architecture

Transductive learning has been successfully utilized along with encoder-decoder structure in many language tasks (**S2SLearningWithNNs_Sutskever**; **LearningPhraseRepresentations_Cho**; **AttentionIsAllYouNeed_Vaswani**). Transduction is first proposed in **LearningByTransduction_Gammer** to counteract with unlabeled data problem. In contrast to supervised learning, transductive learning does not require all data to be labeled, instead it utilizes the clustered behavior of data. Using the gaps between different clusters and a small set of labeled data, transductive learning assigns labels to unlabeled data. Accordingly, Transformer models are transductive models and use encoder-decoder structure to achieve that.
Encoder-decoder structure that takes into account the order of words was proposed in (**LearningPhraseRepresentations_Cho**). This encoder-decoder structure consists of one RNN as encoder and one RNN as decoder. The encoder maps an input sequence to fixed-length vector, and the decoder maps this fixed-length vector to a target sequence. Transformer architecture adopts a similar approach which employs feed-forward and Multi-Head Attention layers in both encoder and decoder which is illustrated in Fig. **??** with N=6 stacks of encoders and decoders.
In order to reduce sequential computation, CNNs have been adopted as building blocks that parallelly compute hidden representations for all input and output positions (**AttentionIsAllYouNeed_Vaswani**). Although aimed to reduce computation, the number of operations to convey information from one random input or output to another increases linealy in ByteNet (**ByteNet_Kalchbrenner**) and logarithmically in ConvS2S (**ConvS2S_Gehring**). Contrary to CNNs, Transformers are able to fix the number of operations by averaging attention-weighed positions which decreases the effective resolution. However, this decrease in resolution is neutralized by the utilization of Multi-Head Attention (**AttentionIsAllYouNeed_Vaswani**). Initially suggested in the decoder of the model proposed in (**NeuralMachineTranslationByJointlyLearning_Bahdanau**), an attention mechanism works similar to human attention; it learns to put more importance on some words that convey the relevant information about the sentence. It does so by means of a context vector that depends on a sequence of *annotations*. An annotation h_i for a word (or embedding) x_i contains information about the complete input sentence but with a focus on the words that are closer to the word x_i. The context vector c_i for word x_i is obtained as a weighted sum of all these annotations h_i:

$$c_i = \sum_{j=1}^{|x|} \alpha_{ij} h_j.$$

The weight α_{ij} is obtained by applying softmax to associated energy e_{ij} which is an output of the alignment model a. The alignment model a is a feed forward neural network that jointly learns with the rest of the system. More precisely, we compute these values as follows:

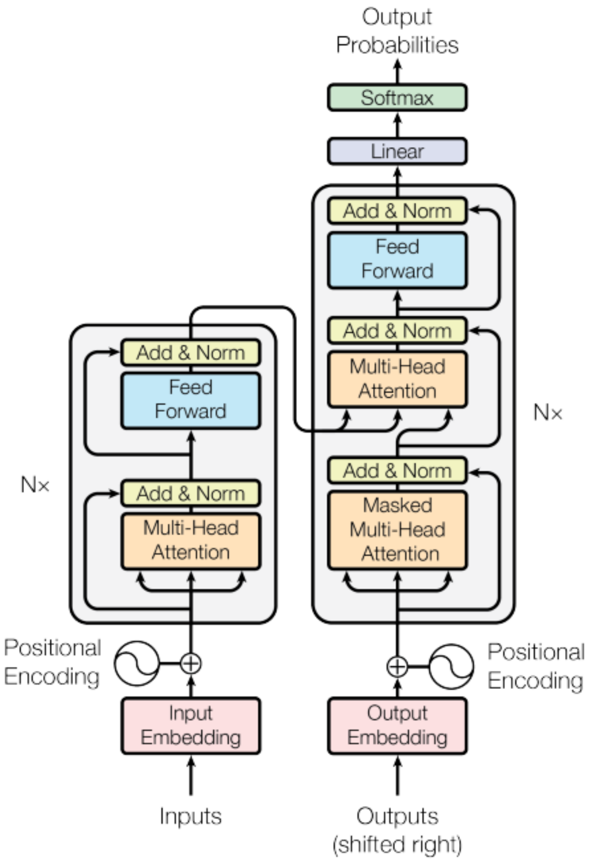

Figure 3.3: The Transformer Model Architecture (N=6). Figure obtained from (**AttentionIsAllYouNeed_Vaswani**)

$$\alpha_{ij} = \frac{e_{ij}}{\sum_{k=1}^{|x|} exp(e_{ik})}$$

where

$$e_{ij} = a(s_{i-1}, h_j)$$

with s_i representing the current and s_{i-1} the previous state of the model (**NeuralMachineTranslationByJoi**
This is called *additive attention*.

For Transformer models, the authors define an attention function as *mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compability function of the query with the corresponding key.* (**AttentionIsAllYouNeed_Vaswani**). The Transformer model employs two different attention mechanisms, namely, *Scaled Dot-Product Attention* and *Multi-Head Attention*. By following the notation in (**AttentionIsAllYouNeed_Vaswani**), we denote that the input for the attention layers are matrices called queries $Q \in \mathbb{R}^{d_{model} x d_k}$, keys $K \in \mathbb{R}^{d_{model} x d_k}$, and values $V \in \mathbb{R}^{d_{model} x d_v}$, with d_{model} being the model dimension. Scaled Dot-Product Attention computes the dot product of all queries $q_i \in Q$ with all keys K and scale the resulting wegihts with $\frac{1}{\sqrt{d_k}}$. After obtaining the softmax of the scaled weights, each weight is multiplied with the correspoding value to obtain attention values.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

The second attention mechanism, Multi-Head Attention, uses multiple attentions each of which uses a different learned linear projection of Q, K, V. Output of each of these attentions are then concatenated to obtain the final result. More precisely, it is computed as follows,

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \ldots, head_h)W^O$$

where each $head_i$ is calculated as,

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

with projection for Q as $W_i^Q \in \mathbb{R}^{d_{model} x d_k}$, K as $W_i^K \in \mathbb{R}^{d_{model} x d_k}$, Q as $W_i^V \in \mathbb{R}^{d_{model} x d_v}$, and lastly, $W^O \in \mathbb{R}^{hd_v x d_{model}}$ (**AttentionIsAllYouNeed_Vaswani**).

We now discuss each part of the Transformer model briefly.

- *Encoder*: Consists of N=6 identical layers. Each of these layers have two sublayers the first of which uses a Multi-Head Attention and layer normalization (**LayerNorm_Ba**) along with a residual connection around. The second sublayer consists of a feed-forward layer and layer normalization as well as a residual connection (**ResidualConnection_He**) around the feed-forward layer (**AttentionIsAllYouNeed_Vasw**

- *Decoder*: Same as the encoder, this part is also composed of N=6 layers. Additional to the previously discussed two sub-layers in encoder, decoder adopts a third sub-layer which computes the attention values over the output of encoder. As it was done for encoder, decoder also utilizes layer normalization at the end of each sub-layer as well as the residual connection (**AttentionIsAllYouNeed_Vaswani**).

Each of the feed forward networks in the sub-layers are position-wise, meanining that they are applied to each position separately and identically. These feed-forward networks use different parameters for each layer. The embeddings are obtained through transductive learning. Lastly, the positional encodings for input embeddings are calculated using sine and cosine functions of different frequencies (**AttentionIsAllYouNeed_Vaswani**). We have summarized the Transformers architecture in order to lay foundations for the model we use, DistilRoBERTa (**DistilBERT_Sanh**; **RoBERTa_Liu**). Next, we initially introduce details of BERT, then RoBERTa, and lastly DistilRoBERTa which is our news content model.

3.1.3 Model, Dataset, Tokenizer Analysis

We employ a fine-tuned version of case-sensitive Transformer model DistilRoBERTa for our task of FND with news content. DistilRoBERTa is a distilled version of *A Robustly Optimized BERT Pretraining Approach* (RoBERTa) (**RoBERTa_Liu**). It uses the same distillation procedure adopted for DistilBERT (**DistilBERT_Sanh**) to *Bidirectional Encoder Representations from Transformers* (BERT) (**BERT_Devlin**). This distillation procedure is referred to as *Knowledge Distillation* and it compresses a model (**ModelCompression_Bucilua**) - the teacher - by means of training a smaller model - the student - to reproduce the same behaviour (**DistillingTheKnowledge_Hinton**). In our case, the teacher is RoBERTa and the student is DistilRoBERTa. First, in order to examine properties of RoBERTa, we discuss BERT in detail.
As the name suggests, the model architecture of BERT is a multi-layer bidirectional Transformer based on the Transformer model. BERT uses BookCorpus (**BookCorpus_Yukun**) and English Wikipedia as training dataset, with two training objectives, *Masked Language Modeling* (MLM) and *Next Sentence Prediction* (NSP). MLM procedure applies the following for each sentence sampled from a document in the cumulative dataset.

- Mask 15% of the tokens.
- In 80% of the cases, replace the masked tokens with [MASK].
- In 10% of the cases, replace the masked tokens with a different random vocabulary token.

- In the remaining 10% of the cases, the masked tokens are left unchanged.

NSP procedure is a binary classification loss that predicts whether two segments (sequences of tokens) are consecutive in the original text. *Positive* and *negative* examples are sampled with equal probability in this process. Positive examples are created with taking the consecutive sentences from the text corpus, whereas negative examples are generated by pairing segments from different documents (**RoBERTa_Liu**).
RoBERTa is an optimized, longer pretrained with longer sequences version of a BERT implementation that uses only MLM as training objective. Contrary to BERT, RoBERTa keeps a dynamic masking pattern that changes in training. It is pretrained on reunion of five datasets (three more datasets than BERT) that size up to 160 gigabytes (GB): BookCorpus (**BookCorpus_Yukun**), English Wikipedia (**EnglishWikipedia_Wiki**), CC-News (**CCNews_Nagel**), OpenWebText (**OpenWebText_Radford**), Stories (**ASimpleMethodForCommons**
RoBERTa tokenizes texts using BPE with a vocabulary size of 50,000 and maximum sequence length (maximum number of tokens) as 512. The beginning and end of each document (news article) is marked with <s> and </s> respectively. With MLM as training objective and Adam (**Adam_Kingma**) as the optimizer, the model reaches better results than BERT. Additionally, it should be noted that since these models are further trained for downstream tasks, thus we refer to training stage as pretraining to avoid any confusion.
DistilRoBERTa has the same general architecture as RoBERTa but the number of layers are reduced by a factor of two, the *token-type embeddings* and the pooler are removed. Then DistilRoBERTa is initialized with layers from the teacher. The distillation is done with very large batches (**DistilBERT_Sanh**). Using RoBERTa as a teacher, the student DistilRoBERTa is pretrained on OpenWebTextCorpus (**OpenWebTextCorpus_Gokaslan**) a reproduction of OpenWebText (**OpenWebText_Radford**).
We employ a fine-tuned version of DistilRoBERTa from Huggingface[1] that is trained on a dataset curated from different sources[2]. Although there exist better datasets and news content models, we opted for this particular model for two reasons. First, most SOTA news content models do not provide their code and dataset to reproduce results. Second, since Transformers are SOTA and this particular trained model provides us with not only the dataset but also with the train/test/validation splits which allows us to analyze its explanation. Now, we analyze the dataset, convey the distribution of labels and discuss potential peculiarities. Then we talk about the performance of the model, and reason about it.
This dataset includes

[1] https://huggingface.co/GonzaloA/distilroberta-base-finetuned-fakeNews

[2] https://huggingface.co/datasets/GonzaloA/fake$_{n}ews$

3.2 Social Context Models

Talk about models that incorporate social context, spatiotemporal information and other context with text data. Can be any kind of model.

3.2.1 Notation and Definitions

3.2.2 Geometric Deep Learning

Talk about Graph Neural Networks

3.2.3 Dataset

FakeNewsNet, UPFD, explain the dataset, no of edges/nodes. Which models use this dataset,

3.2.4 Models

SAGE GNN UPFD GCNFN

3.3 Early Fake News Detection and Model Aging

4 Explainability of Fake News Detection Models

4.1 Explanation Techiques

4.1.1 SHAP, DeepSHAP

4.1.2 GNNExplainer

4.1.3 Explainability vs. Explanation

4.2 Content Based Fake News Detection Models

4.2.1 Explaining Content Based Fake News Detection Models

4.2.2 Introducing Unseen Data

4.2.3 Results

4.3 Content and Social Feature Based Fake News Detection Models

4.3.1 Explaining Content and Social Feature Based Fake News Detection Models

4.3.2 Introducing Unseen Data

4.3.3 Results

5 Conclusion

List of Figures

List of Tables