UKRAINIAN CATHOLIC UNIVERSITY

BACHELOR THESIS

Fake News Epidemic Model Development

Author:
Pavlo YASINOVSKYI

Supervisor:
Dr. Jaroslav ILNYTSKYI

*A thesis submitted in fulfillment of the requirements
for the degree of Bachelor of Science*

in the

Department of Computer Sciences and Information Technologies
Faculty of Applied Sciences

Lviv 2023

Declaration of Authorship

I, Pavlo YASINOVSKYI, declare that this thesis titled, “Fake News Epidemic Model Development” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ________________________________

Date: ________________________________

"Truth is the first casualty of war."

Aeschylus

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Bachelor of Science

Fake News Epidemic Model Development

by Pavlo YASINOVSKYI

Abstract

As the commonness of fake news continues to grow, understanding its spread and impact on social networks is of paramount importance. This paper presents a C++ implementation of a previously described SIR-like Susceptible-Infected-Fact-Checker (SIFC) model to study the dynamics of fake news spread. Our solution provides significant performance enabling researchers to simulate massive social networks and examine how different variables affect the propagation of fake news. In order to assess how accurately the false news epidemic in a social network is depicted, we compare our model with real-world data. Our research provides useful information that authorities, social media managers, and users may utilize to build policies that will stop the spread of fake news and foster a more positive online community.

Acknowledgements

I want to express my gratitude to Jaroslav Ilnytskyi for suggesting an interesting topic and being my mentor throughout the research, Oleg Farenyuk, and the Armed Forces of Ukraine, who made this paper possible.

Contents

List of Figures

List of Tables

Dedicated to the Defence Intelligence of Ukraine and all the people who defend the independence of Ukraine.

Chapter 1

Introduction

1.1 Background on Fake News and its Impact on Society: The Case of the russian-Ukrainian War

In the digital era, fake news is more common than ever, thanks to social media's explosive growth and the ease with which information can be shared between platforms. This false or misleading information may harm both public opinion and political dialogue. Fake news has also been known to spark actual hostilities in some circumstances, as is the case during the russian-Ukrainian War.

The start of the russian-Ukrainian War in 2014 perfectly illustrates how fake news can serve as a powerful weapon in information warfare. Several agents, including state authorities, have been spreading false information in an attempt to influence public opinion and advance their own agendas. As a result, the public's perception of events is skewed, and the information landscape is complicated and frequently confused.

Disinformation efforts aimed at influencing public opinion in Ukraine and abroad have been one of the primary tactics used by russians throughout the War. Through these activities, russia's military actions are being made to appear more justified while also undermining the legitimacy of the Ukrainian government. The fake news included fabrications of atrocities committed by Ukrainian forces against russian-speaking ethnic minorities and untrue assertions that the West intervened in the conflict.

Such false information has a significant impact on society. While Ukraine has built some resilience against russian propaganda, the spread of fake news in countries not directly involved in the War has contributed to a polarized discourse around the events, with different sides accusing one another of manipulation and deception. This hinders the unification of the free world to overcome a common enemy.

In summary, due to its ability to sway public opinion, undercut democratic processes, and even contribute to actual conflicts, the growing number of fake news on social media is a serious problem. Understanding the mechanisms of fake news's propagation on social media and creating practical strategies to lessen its effects is thus of utmost importance.

1.2 Practical Value of Modeling and Simulating Fake News Spread in Social Networks

The mechanics underlying the propagation of misleading information and the efficacy of various defenses can be better understood by modeling and simulating the phenomenon.

The main components of this process are outlined below:

1. **Identifying the key factors in the spread of fake news:** These elements may include the network's structure, the impact of influencers and bots, the importance of social proof, and the type of shared content. By understanding these elements, researchers may build focused strategies to stop the spread of fake news and develop a deeper understanding of the mechanisms contributing to its spread.

2. **Evaluating the effectiveness of countermeasures:** For instance, researchers can examine the efficacy of fact-checking programs, the function of algorithmic platform provider interventions, or the effect of public awareness campaigns. Policymakers can decide which tactics to use to stop the spread of fake news most successfully by examining the efficacy of various countermeasures.

3. **Developing a communication policy and platform:** For stakeholders, understanding the dynamics of the spread of disinformation can form the basis for developing regulations and recommendations aimed at minimizing its impact on society. For platform developers, the modeling results can serve as a basis for developing algorithms and functions to disseminate accurate information while limiting access to false content.

4. **Enhancing public awareness and media literacy:** By modeling and simulating the spread of fake news on social media, researchers can also help raise public awareness of the problem and promote media literacy. People can become more discerning online content consumers and better able to recognize and reject misleading information by increasing public understanding of the variables that contribute to the spread of disinformation.

Therefore, in order to understand the negative effects of disinformation and counteract it, it is crucial to analyze and simulate how fake news spreads on social media. Researchers and stakeholders can collaborate to build a better-informed and less divisive online environment by identifying critical components in the propagation process, assessing the efficacy of remedies, and shaping platform policy and design.

1.3 Purpose of the Current Study: Developing an Efficient C++ Model and Validating It on Real Data

This research aims to implement the referenced SBFC model [6] in C++, a programming language known for its performance. The main motivation for this is to improve its computational efficiency, making it more suitable for large-scale social media analysis and real-world applications. In addition to evaluating the new version's performance, we compare it with real data and discuss its applicability in practical scenarios.

Chapter 2

Literature Overview

2.1 Previous Research on Modeling Fake News Epidemics on Social Media

Both researchers and politicians have paid close attention to the issue of fake news propagating on social media. In order to better understand the mechanisms underlying this issue and create plans to lessen its effects, a number of researches have been carried out to model and simulate the propagation of fake news. This section gives a summary of earlier research in the field and highlights several methodologies and tactics used to simulate false news epidemics on social media.

2.1.1 Graph Model Analysis of Fake News Spread

We begin our overview with [7], where the authors thoroughly analyze existing literature, focusing on four key aspects: agent behavior, propagation patterns, transition patterns between user states, and network topology.

They explain the graph model used to study the spread of fake news in social networks. The graph model consists of nodes (individuals) and edges that connect them. Nodes have properties such as state (e.g., unaware, spreader, carrier, recovered). Some models incorporate node polarity (e.g., positive, negative, neutral) that affects the user′s interaction with the information they receive.

The authors differentiate nodes into two categories: transmitters and receivers. Transmitters disseminate information and have four properties: reaction time, persistence, authority, and sensitivity. They determine the behavior of the transmitter and can be used to model different users.

In receivers, the defining characteristic is their receptivity to information. Factors such as attitude, the number of messages received, and the authority of the source affect how likely they are to change their opinion or state after interacting with the information.

Speaking of the user′s attitude to information, not all models incorporate this factor. Therefore, such a wide range of dimensions allows us to come up with various interactions that create opportunities for the introduction of new models in the literature.

2.1.2 Types of Network Topologies

Network topology plays a crucial role in information dissemination as different structures can affect the spread of fake news. This section discusses the most popular topologies and their impact on the propagation process:

1. **Random network:** A network in which nodes are connected randomly according to a probability distribution. Erdos and Renyi provided a pioneering example with their model [3].

2. **Scale-free network:** A particular case of a random network with a power law degree distribution. These networks are widely used in the literature, and studies have shown that rumors spread more easily in scale-free networks than in random networks.

3. **Heterogeneous network:** A network in which nodes are categorized based on function and utility. These networks consider different degrees of connectivity and are more realistic when modeling modern social networks.

4. **Dense and sparse networks:** Dense networks have a large number of neighbors (edges) for each node, while sparse networks have fewer edges. Dense networks are less vulnerable to social fragmentation and have a greater flow of information to more people.

5. **Correlated networks:** These networks display degree correlations and can be either assortative or disassortative. The speed of propagation and the final size of the rumor may be different in correlated networks compared to uncorrelated networks.

6. **Hierarchical networks:** These are history-dependent networks in which members change their behavior when interacting with people who joined earlier. The spread of an epidemic in a hierarchical social network depends on the clustering coefficient and the incubation period.

2.1.3 Epidemiological Models and Their Application in the Study of the Spread of Fake News

The authors discuss several mathematical models of the spread of fake news focusing on epidemiological models due to their similarity to the spread of infectious diseases [7]. One of their subtypes is used in this paper. Epidemiological models can be deterministic (compartmentalized) or stochastic.

Deterministic models assume that the population is divided into smaller classes (or compartments) representing different stages of the epidemic, and the transitions between these stages are deterministic. Stochastic models, on the other hand, assume probabilistic state transitions and take into account variations in input variables.

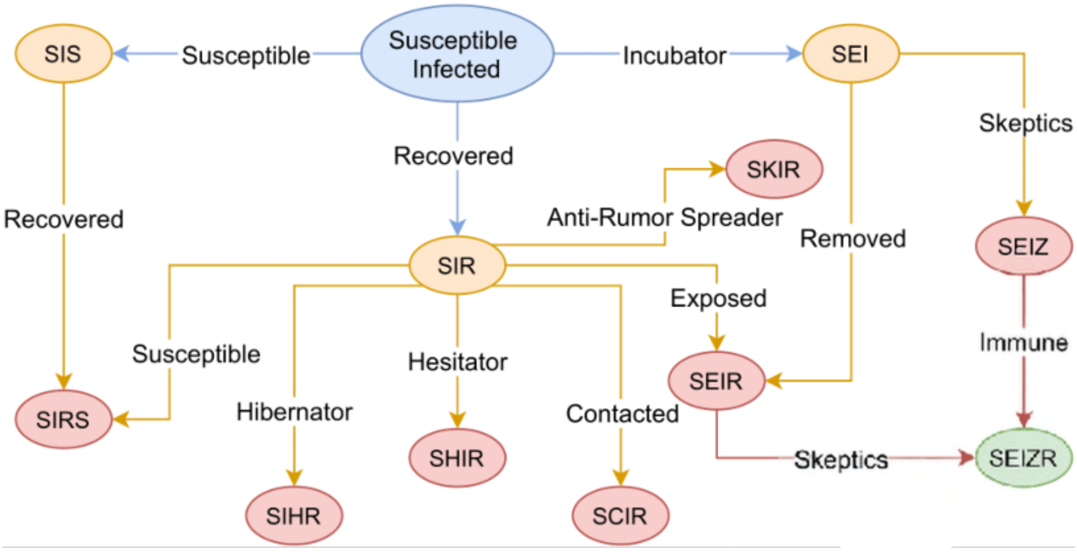


FIGURE 2.1: Epidemiological Models built on top of the SI model (Raponi, 2022, p. 9)

Various epidemiological models are presented, including the five most representative ones:

1. **SIS (Susceptible-Infected-Susceptible) model:** People can only be in one of two states according to the SIS model: susceptible (S) or infected (I). Susceptible individuals are those who have not been exposed to fake news, whereas infected people have been exposed to it and are spreading it. At one point, they cease disseminating fake news. Still, they are vulnerable again the next time they are exposed to it because they lack immunity. This model is appropriate for researching the dynamics of fake news that spreads continuously through society without lasting effects on people's opinions.

2. **SIR (Susceptible-Infected-Recovered) model:** This model differs from the previous one in that those who stop spreading fake news and develop immunity to its effects transition to recovered (R) state. This model helps to understand how fake news spreads when people become resistant to false information after being exposed to it, whether through fact-checking or other preventive actions.

3. **SIRS (Susceptible-Infected-Recovered-Susceptible) model:** As in the SIR model, susceptible individuals can become infected and recover. However, in the SIRS model, people who have recovered can lose their immunity over time and become vulnerable to fake news again. This model applies to situations where resistance to disinformation is temporary, and after a certain period, people may be infected again.

4. **SKIR (Susceptible-Knowledgeable-Infected-Recovered) model:** The SKIR model introduces a new state called "knowledgeable" (K) to represent individuals who are aware of fake news but are not infected and do not spread it. A variation of the SKIR model, SIFC (Susceptible, Infected, Fact Checker), emphasizes the fact-checkers role in information dissemination. Fact-checkers can identify and correct misinformation, potentially reducing the spread of fake news or rumors and helping people return to normal.

5. **SEI (Susceptible-Exposed-Infected) model:** The SEI model is a variant of the classic SIR model. Exposed individuals (E) have encountered the fake news but are not yet actively spreading it, representing an "incubation" period during which they consider the information.

2.1.4 Why Incubation Time in Fake News Spread Models Can Be Neglected In the Modern Digital Landscape

Literature suggests that models built on top of SEI allow for a more nuanced representation of the propagation process by incorporating the concept of incubation time between exposure and active spreading [7].

However, it can be argued that the significance of incubation periods might be diminishing in the current fast-paced digital landscape. Here are some reasons why incubation might be considered negligible in the context of modern information dissemination:

1. **Instant sharing:** Social media platforms have made it incredibly easy for users to share information with a simple click or tap. This ease of sharing can significantly reduce the incubation period, as people might impulsively share information without thoroughly evaluating its credibility.

2. **Virality:** The viral nature of social media can lead to the rapid spread of fake news, with users quickly sharing and resharing content. In these cases, the time between exposure to misinformation and the decision to share might be so short that considering incubation periods becomes less meaningful.

3. **FOMO (Fear of Missing Out):** The desire to stay informed and up-to-date with the latest news might compel individuals to share information quickly without taking the time to process or verify its validity. This urgency can result in significantly shortened incubation periods.

4. **Algorithmic amplification:** Social media algorithms often prioritize engagement, rapidly promoting sensational and controversial content, including fake news. These algorithms can accelerate the spread of misinformation, effectively bypassing the incubation period.

5. **Echo chambers:** The tendency for people to cluster around like-minded individuals on social media can create echo chambers, where misinformation is reinforced and quickly spread among the group. The incubation period might be shortened even more in these environments, as individuals are more likely to accept and share information that aligns with their existing beliefs.

Therefore, this paper assumes that the incubation time can be considered negligible.

2.2 Limitations and shortcomings of existing models

Despite the progress in modeling and simulating fake news epidemics on social media, some models still need to improve their ability to capture the complex dynamics of fake news spread accurately. This section covers the main limitations and shortcomings found in previous models:

1. **Synchronous dynamics:** Many existing models assume that every agent in the network has synchronous access, meaning that they all interact with one another simultaneously. This method does not effectively reflect the asynchronous nature of real-world social networks, where people access the network at various times and at various rates.

2. **Uniform trust:** A small number of models ignore the varying degrees of trust between nodes, assuming that every connection has the same impact. This simplification ignores the effect of trust on information spread since people are more likely to believe and share information from reliable sources.

3. **Constant fake news engagement:** Continuous exposure to fake news: Many models assume constant exposure to fake news, ignoring the fact that interest in fake news tends to decline over time.

4. **Lack of realism in agent behavior:** The proper choice of agent behavior: Just as the behavior of agents should not be too simple, leading to simulations that are unduly idealistic, it should not be too complex, as this will make it challenging to identify the major elements that contribute to the spread of fake news.

5. **Insufficient treatment of influencers, bots, and fact-checkers:** Influencers, bots, and fact-checkers are not adequately considered in many studies, which contributes to the spread of false information. Modern models need to incorporate these individuals who fuel epidemics of false news on social media.

6. **Limited validation with real data:** Some models' applicability and generalizability in the social media ecosystem are in doubt since they have yet to undergo comprehensive testing and validation using real data.

Addressing these limitations and shortcomings is essential to developing more accurate and comprehensive models of fake news epidemics on social media. By incorporating more realistic characteristics such as temporal dynamics, uneven trust, declining interest in news, and diverse agent behavior, future models can provide a more reliable understanding of the factors that influence the spread of fake news and help develop effective strategies to mitigate its impact on society.

2.3 The SBFC Model: Improvements and Features

The SBFC model is an agent-based simulation system designed to study the dynamics of fake news spreading in social networks in a more realistic manner. It incorporates such features as time dynamics, node-dependent attributes, weighted network edges, and the roles of influencers, bots, and eternal fact-checkers. Below is a detailed description of the model, including the algorithms and constants used:

1. **SBFC model:** The model takes its idea from the SIR framework and divides the nodes (agents) in a network into three states: susceptible (S), believer (B), and fact-checker (FC). Susceptible nodes have not yet encountered the fake news, believer nodes have encountered and believe the fake news, and fact-checker nodes have encountered and started debunking the fake news.

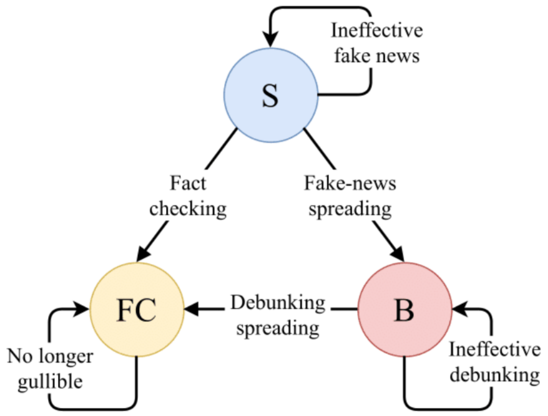


FIGURE 2.2: State Transition Diagram of Regular Users and Influencers (Lotito, 2021, p. 10)

2. **Time dynamics:** The model incorporates time dynamics by assigning each agent an independent access time to the network representing the agent's social media usage frequency. The agents' access times are drawn from an exponential distribution with a mean access rate parameter. This allows the simulation to model asynchronous interactions among agents, reflecting real-world social network usage patterns.

3. **News engagement decay:** The model includes a decay factor to represent the decrease in engagement with a fake news item over time. This decay factor is modeled using an exponential decay function, where the engagement level at a given time is multiplied by a decay constant.

4. **Weighted networks and non-uniform trust:** The model uses directed, weighted network edges to represent the varying levels of trust between nodes. The weights are assigned based on the roles of the nodes (e.g., common users, influencers, bots) and represent the level of trust or influence they have over their connected nodes. This allows the model to capture the impact of trust on information flow and fake news spreading more accurately.

5. **Influencers, bots, and fact-checkers:** The model explicitly includes the roles of influencers, bots, and eternal fact-checkers in the fake news spreading process. Influencers have a higher degree of connectivity and impact on their followers, while bots are programmed to spread fake news more frequently. Eternal fact-checkers are agents that never become believers and always work to debunk fake news. The model assigns these roles based on predefined probabilities and distributions.

6. **Agent behaviors:** The model introduces realistic agent behaviors by sampling node attributes and behaviors from statistical distributions. This approach ensures a diverse range of agent characteristics, better reflecting real-world social network users.

7. **Simulation algorithm:** The model uses a discrete event simulation approach, where events are processed in chronological order. At each step, an agent is selected based on its access time, and the agent interacts with its connected nodes by either spreading fake news or debunking it, depending on its current state. The simulation continues until a maximum number of iterations is met.

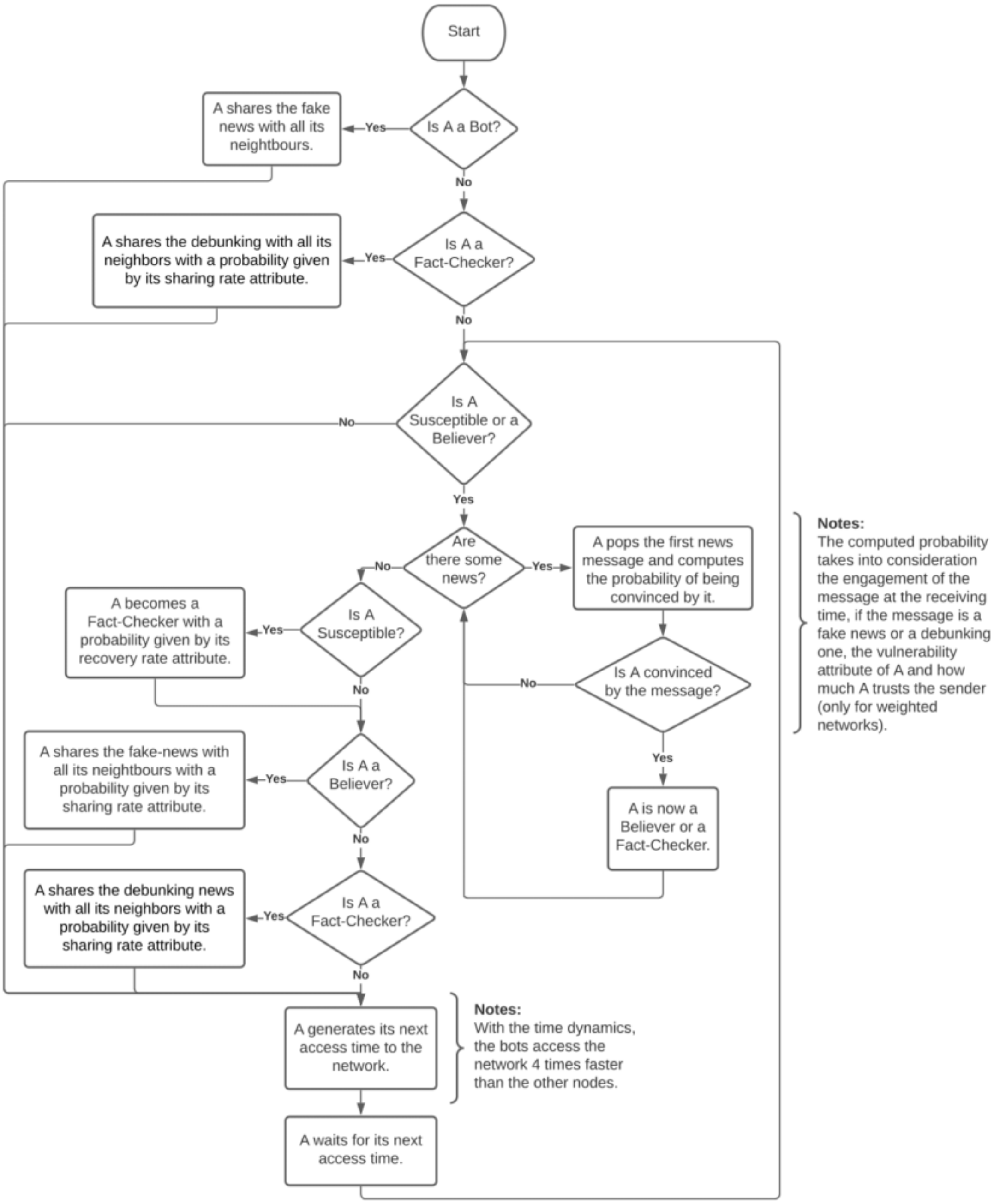

Start
Is A a Bot?
Yes
A shares the fake news with all its neightbours.
No
Is A a Fact-Checker?
Yes
A shares the debunking with all its neighbors with a probability given by its sharing rate attribute.
No
Is A Susceptible or a Believer?
No
Yes
Are there some news?
Yes
A pops the first news message and computes the probability of being convinced by it.
Notes:
The computed probability takes into consideration the engagement of the message at the receiving time, if the message is a fake news or a debunking one, the vulnerability attribute of A and how much A trusts the sender (only for weighted networks).
No
Is A Susceptible?
Yes
A becomes a Fact-Checker with a probability given by its recovery rate attribute.
Is A convinced by the message?
No
Yes
A is now a Believer or a Fact-Checker.
No
Is A a Believer?
Yes
A shares the fake-news with all its neightbours with a probability given by its sharing rate attribute.
No
Is A a Fact-Checker?
Yes
A shares the debunking news with all its neighbors with a probability given by its sharing rate attribute.
No
A generates its next access time to the network.
Notes:
With the time dynamics, the bots access the network 4 times faster than the other nodes.
A waits for its next access time.

FIGURE 2.3: Block scheme of the simulation algorithm illustrating the step-by-step process of news propagation within the network (Lotito, 2021, p. 11)

To be consistent with the epidemiological convention, this paper refers to this model as SIFC from now on.

Chapter 3

Methodology

The C++ version of the SIFC model was created from scratch according to how it was described in the study. The source code is available at [1]. This section outlines the crucial steps in its implementation and optimization process.

3.1 MVC Design Pattern

In our C++ implementation of the fake news simulation model, we have employed the Model-View-Controller (MVC) design pattern to separate the program's business logic from its graphical user interface (GUI). In addition to leading to a more efficient and modular program structure, this approach allows us to reuse the code in a separate version to generate data without the overhead of a GUI.

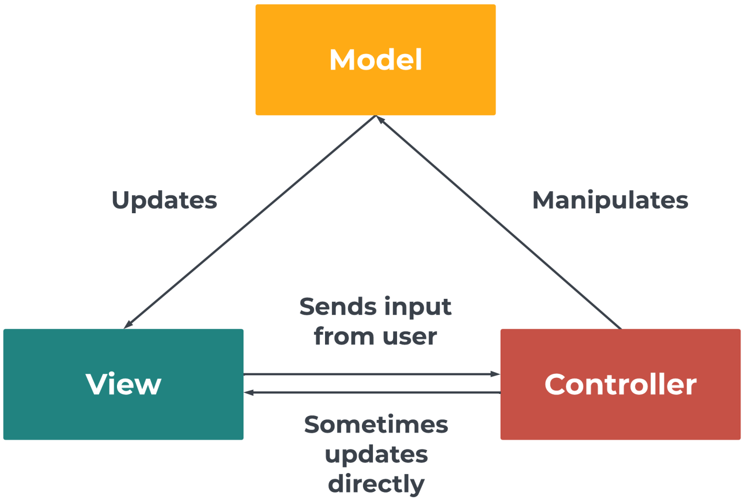


FIGURE 3.1: Model-View-Controller (MVC) diagram showcasing the architecture of the application and the interaction between different components.

1. **Model:** The Model component represents the fake news simulation's core business logic and data structures (e.g., a graph, nodes). It contains the algorithms for simulating the spread of fake news, the network structure, and the properties of agents.

2. **View:** The View component displays the simulation data to the user. In our case, it is implemented as a GUI via the Qt framework. However, by separating it from the Model, we can run the simulation without any visualization at all, reducing the computational overhead.

3. **Controller:** The Controller component manages the user input and communicates with the Model and View components to orchestrate the simulation process. It acts as an intermediary between the user interface (View) and the underlying data structures and algorithms (Model).

Despite this and other improvements made, the program structure is inherently non-parallelizable. This limitation arises from the sequential nature of user access to the network, which is a key aspect of the simulation model.

3.2 Network Generation

The network model is built based on the following properties:

1. **Geographical proximity:** Users are more likely to be connected if they live nearby, and we generate coordinates for the nodes using a uniform distribution within a square of side 1. The distance between nodes is calculated using the Euclidean distance formula, and the connection between nodes is established if the distance is below the 0.03 threshold.

2. **Attributes' proximity:** Each node has a set of five "interest" attributes generated using a truncated Gaussian distribution. These attributes are used to model connections between agents based on their interests, creating connections in the attribute domain instead of relying solely on geographical proximity. The threshold for creating a link between nodes is the same as the geographical threshold (0.03).

3. **Randomness:** To introduce some randomness in the network generation process, edges that satisfy the geographical and attribute proximity criteria are removed from the graph with a probability of 50%.

Nodes in the network also have three parameters that affect their behavior in the simulation: vulnerability, sharing rate, and recovery rate. These parameters are assigned randomly to each node using a truncated Gaussian distribution.

Name	Distribution	Description
State	$\{S, B, FC\}$	State of the node
Vulnerability	$\mathcal{N}(0.5, 0.2^2, 0, 1)$	Tendency to follow opinions
Sharing rate	$\mathcal{N}(0.5, 0.2^2, 0, 1)$	Tendency to share the news
Recovery rate	$\mathcal{N}(0.2, 0.2^2, 0, 1)$	Tendency to do fact-checking
Interest attributes	$\mathcal{N}(0, 0.4^2, -1, 1)$	Connect nodes based on their interests
Coordinates	$\mathcal{U}(0, 1)$	Geographical position of the node

TABLE 3.1: Node Attributes

After generating the network with common nodes and creating edges based on the abovementioned rules, we enrich the network with influencers and bots. Influencers have a higher threshold for creating out-edges based on geographical and attribute proximity, leading to more out-connections than in-connections. Conversely, bots are connected to other nodes randomly to attain a target population coverage rate of 2%.

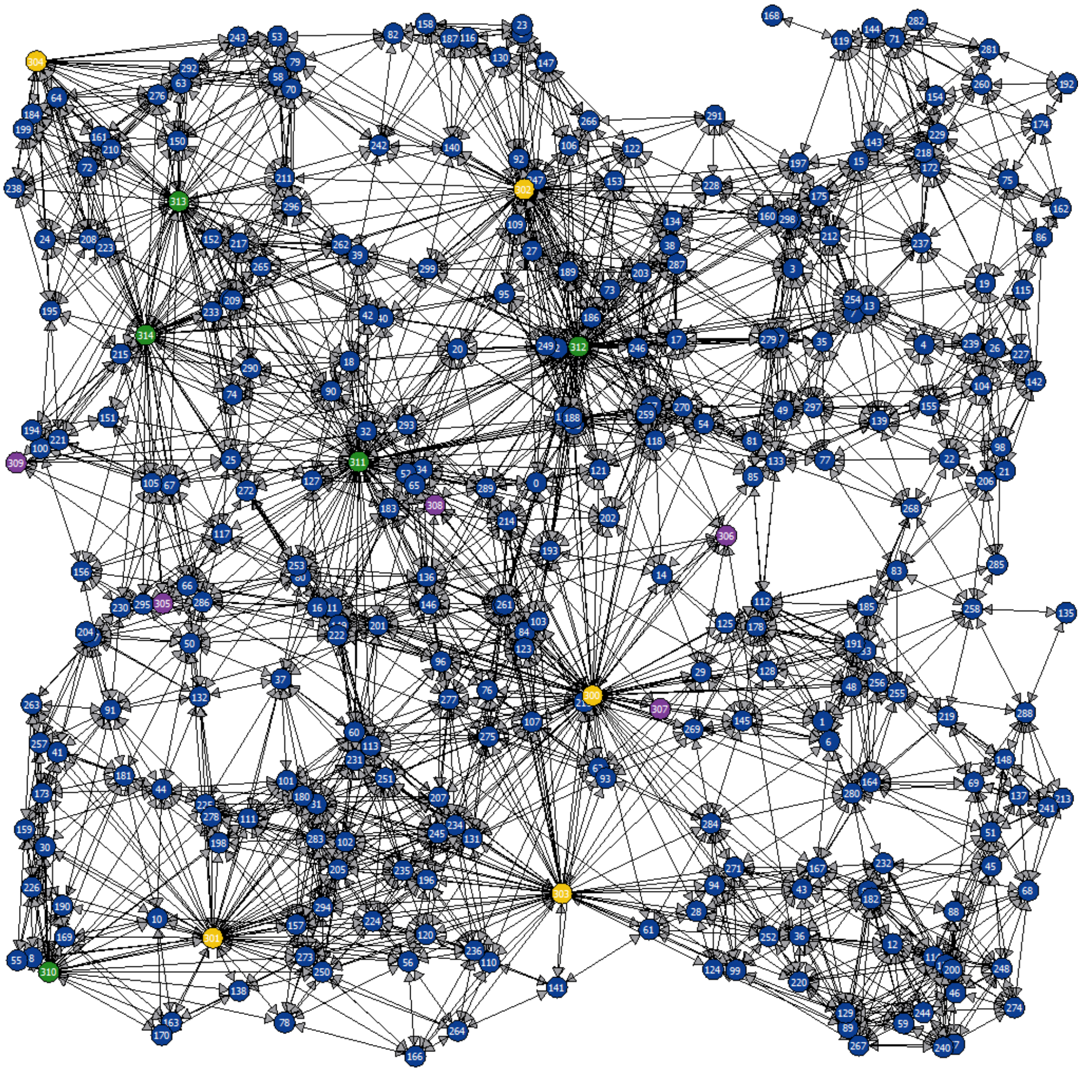


FIGURE 3.2: An example of a generated network. For better clarity, we set the number of regular users (blue) to 300, Influencers (yellow) to 5, Bots (purple) to 3, and Fack-Checkers (green) to 3. We also increased the geographic and attribute thresholds to 0.1 and 0.05 respectively

The C++ implementation also logs information such as the node's ingoing and outgoing edge distribution and the number of susceptible, infected, and fact-checker agents at different timestamps. This information can be used for further analysis and comparisons with real-world datasets.

3.3 Loss of Interest in News Over Time

The model includes an engagement coefficient representing the typical loss of interest in a news item over time. This engagement coefficient varies over time according to the differential equation $\frac{\partial E}{\partial t} = -\gamma E$, where γ is the decay constant.

The solution to this equation, $E(t) = E_0 e^{-\gamma t}$, represents an exponential decay of interest. Here, E_0 is the initial engagement at time $t = 0$, which dictates the initial virality of the news, and γ, the constant engagement decay factor, determines the speed at which the population loses interest in a news item.

The initial engagement E_0 allows to differentiate between the levels of initial engagement expected for different types of messages. For example, E_0 is set for fake news as 1 and for debunking messages as $0.1E_0$ for fake news. This means that debunking messages have a 10-fold lower probability of converting a believer or susceptible individual into a fact-checker than a susceptible individual becoming a believer.

This reflects the confirmation bias behavior, where individuals are more likely to believe in fake news confirming their beliefs than to trust debunking messages. It also acknowledges that scientific news items often fail to reach and influence people.

Furthermore, $\gamma = \frac{2}{T_{\text{sim}}}$, where T_{sim} is the simulation time. This ensures that even by the end of the simulation, fake news still induces some small, non-zero engagement. This value helps simulate the lifespan of fake news, from its peak engagement power to its lowest. At a given time t, $E(t)$ serves as a multiplicative factor that influences the probability of successfully infecting an individual at time t.

3.4 Comparison of the Logged Data With the Real-World Data

We compare the node degree distribution of our model we logged before against a real-world Digg 2009 dataset [4], just as we measure performance growth compared to the Python version.

The Digg 2009 dataset is a snapshot of the Digg social network, which includes user relationships (friendships) and user activities, such as submitting and voting on stories. Despite its age, it is a good representation of the nature of users' connections in social networks.

In order to show how the dynamics of fake news spreading on social media in our model differs from the real world, we compare the number of nodes switching between SIFC (Susceptible-Infected-Fact-checker) states over time in our simulation with the PHEME dataset [5].

The PHEME dataset is a collection of threads discussing real-world events (rumours and non-rumours). Each thread in the dataset is categorized as either a rumour or non-rumour and has associated metadata including a 'misinformation' and 'true' label for indicating the veracity of the news. We wrote a script to traverse the dataset and print the statistics for each event, showing the number of true and fake news items [2].

Event	True News	Fake News
charliehebdo-all-rnr-threads	1814	116
sydneysiege-all-rnr-threads	1081	86
ferguson-all-rnr-threads	869	8
ottawashooting-all-rnr-threads	749	72
germanwings-crash-all-rnr-threads	325	111
prince-toronto-all-rnr-threads	4	222
gurlitt-all-rnr-threads	136	0
putinmissing-all-rnr-threads	112	9
ebola-essien-all-rnr-threads	0	14

TABLE 3.2: PHEME Dataset: Count of Fake and True News per Event

Based on the statistics produced by the script, we identify the event with the most balanced ratio of true to fake news for further analysis. In this case, the event is "germanwings-crash-all-rnr-threads", with 325 true news items against 111 fake news items.

To perform the comparison, we further preprocess the PHEME dataset to obtain the time series of the number of nodes transitioning between SIFC states. We then compare this real-world data with the results obtained from our simulation, focusing on the temporal dynamics of the transitions between the susceptible, infected, and fact-checker states.

Chapter 4

Experiments

In this chapter, we present the experiments conducted to discuss the accuracy and performance of our C++ implementation of the fake news spread model. We focus on various aspects of the simulation, comparing the properties of the generated network (e.g., edge distribution) with real datasets and analyzing the dynamics of the susceptible, infected, and fact-checker agents.

4.1 Comparison of Node Degree Distribution with Digg 2009 Friends Dataset

We compared the node degree distribution of our model against a real-world Digg 2009 Friends dataset within this chapter.

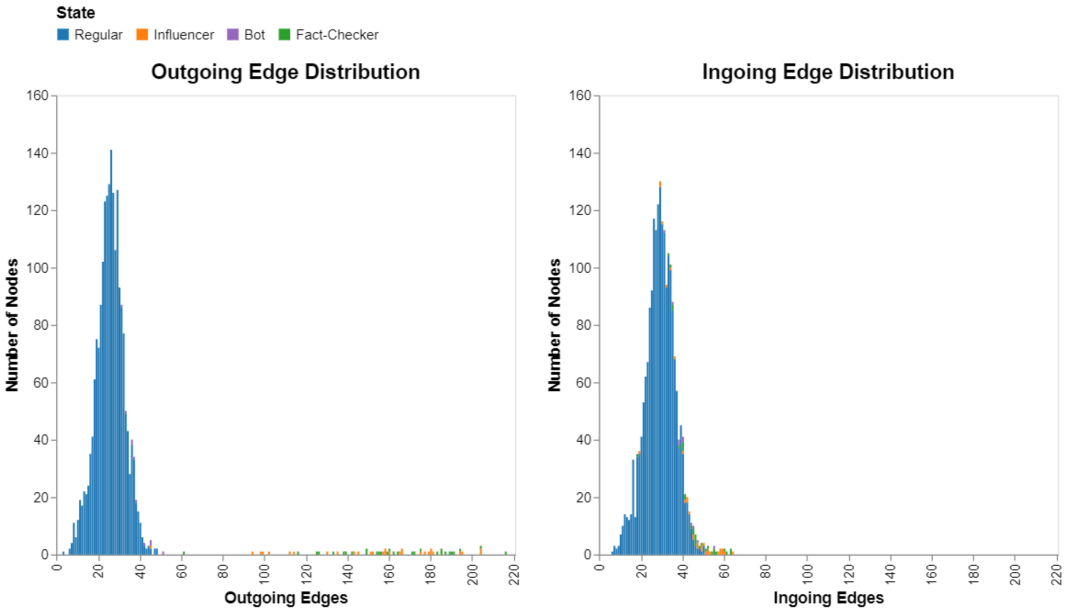


FIGURE 4.1: Edge distribution with states. In this example we set the number of regular users to 2000, Influencers to 30, Bots to 30, and initial Fact-checkers to 30

Our simulated network primarily follows a pattern indicative of a random network, where most nodes have approximately the same degree, and the distribution peaks around an average value. This pattern is a result of our network generation process, which is based on geographical and attribute proximity, with an added element of randomness.

However, the distribution also reveals anomalies, particularly visible on the right side of the outgoing edge distribution edge chart, representing nodes with a high degree. These nodes correspond to the influencers and initial fact-checkers in our network. Their high degree is due to the larger number of outgoing edges, resulting from their unique role in the network – they are responsible for disseminating information to a larger audience.

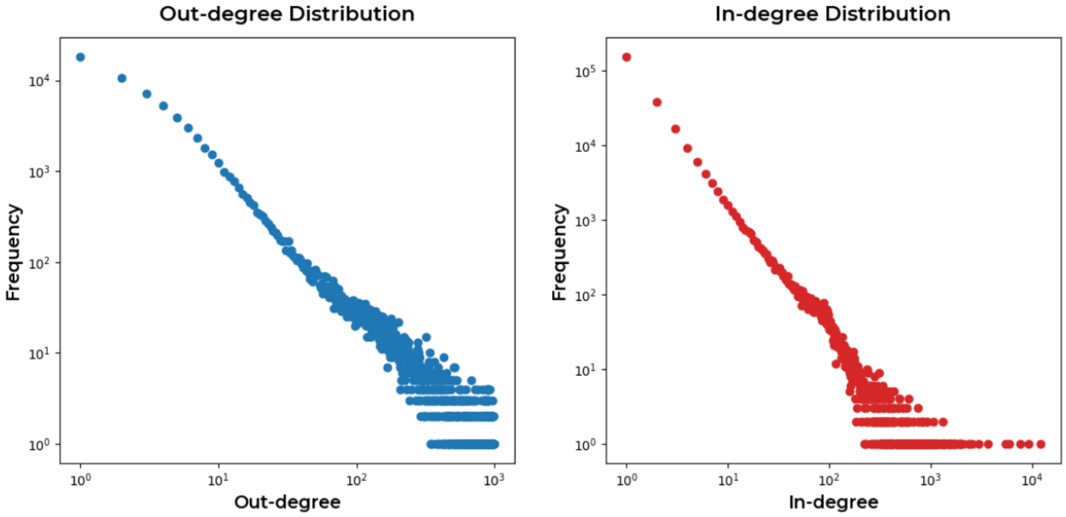


FIGURE 4.2: Digg 2009 Degree Distribution

In contrast, the Digg 2009 Friends Dataset's degree distribution follows a scale-free network pattern. It is the kind of network that is typically seen in real-world social networks. Its defining feature is a power-law distribution, with the majority of nodes having few connections and a small number of nodes, frequently referred to as "hubs," having many connections. In a scale-free network, the hub nodes accelerate the spread of information, potentially leading to faster and broader dissemination of fake news.

Comparing the two distributions, it is apparent that our generated network does not fully replicate the scale-free property of the Digg 2009 network. However, considering our configuration a simplification of real-world network topologies, it can help isolate the effects of other factors, such as network weights, node attributes, and time dynamics, on the spreading of fake news. Additionally, our random network can serve as a helpful baseline for comparing the results of our simulations with other network structures, such as scale-free networks.

4.2 Evaluation of the Impact of Network Configuration on Fake News Spread

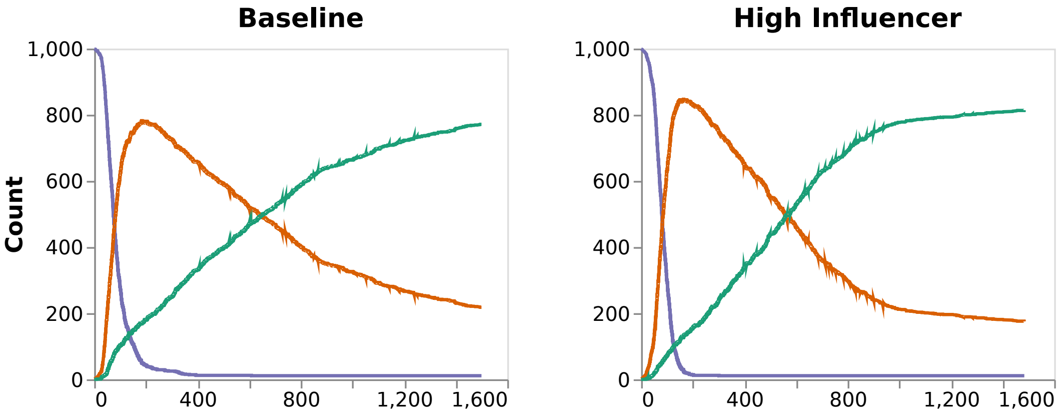
In this chapter, we comprehensively analyze how variations in the numbers of users, bots, influencers, and fact-checkers affect the spread and impact of fake news within the network. We ran a number of simulations with different combinations of agents in order to conduct this analysis.

The following table outlines the configurations tested in the simulations:

Configuration Name	Regular Users	Influencers	Bots	Fact-checkers
Baseline	970	10	10	10
High Influencer	900	80	10	10
High Bot	900	10	80	10
High Fact-checker	900	10	10	80
Mixed 1	850	50	50	50
Mixed 2	850	10	50	90

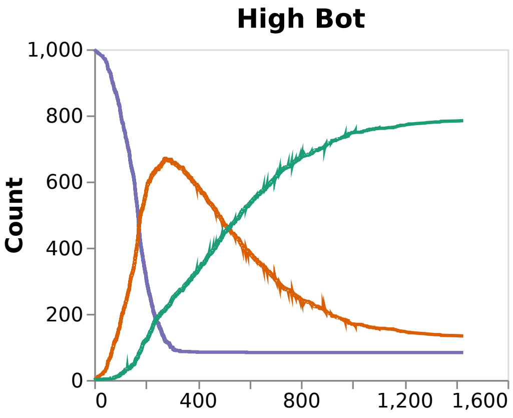
TABLE 4.1: Network Configurations for SIFC Dynamics Analysis

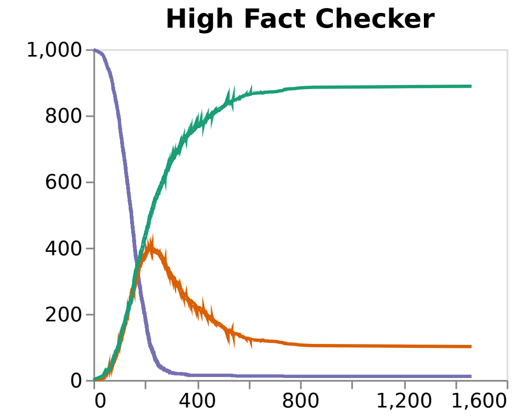
Next, we present the results of these simulations graphically and analyze the outcomes, shedding light on the dynamics of fake news spread under different network configurations.



1. **Baseline Configuration:** This configuration serves as our control group, against which we compare the other setups. It reached a peak of 782 infected users at time 184.15.

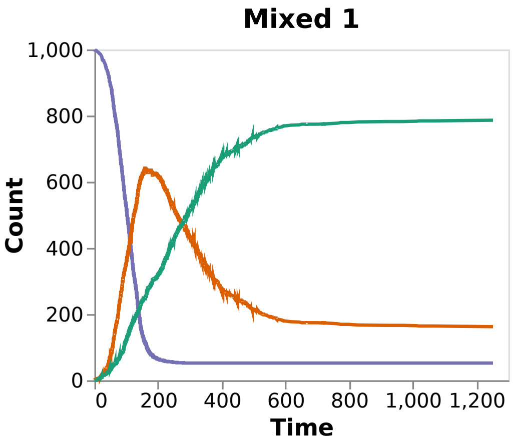
2. **High Influencer Configuration:** The increase in influencers led to a slightly higher peak of infected users (847), which also occurred slightly earlier (time 164.228). This suggests that influencers play the role of accelerators in both infection and recovery. Their high number of outgoing connections allows for a broader and faster dissemination of information, whether accurate or not.

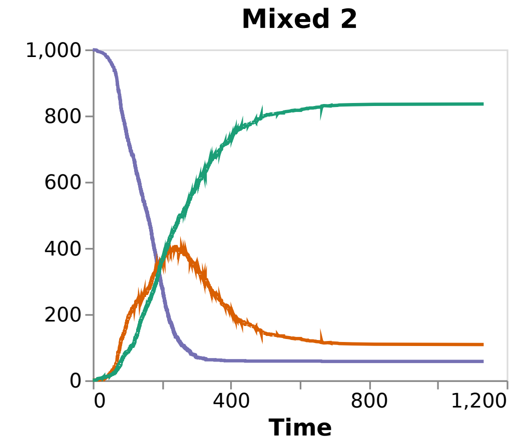

High Bot


High Fact Checker

3. **High Bot Configuration:** Interestingly, the number of infected users at peak (669) was lower than the baseline but occurred significantly later (time 273). This suggests that while bots can spread misinformation, their impact may be less potent than that of influencers, possibly due to their random connections not being as effective.

4. **High Fact-Checker Configuration:** The increase in fact-checkers led to a significant decrease in the peak number of infected users (401), which occurred later (time 221.446). This highlights the importance of fact-checking in mitigating the spread of fake news.


Mixed 1


Mixed 2

5. **Mixed Configuration 1:** The peak number of infected users (641) was lower than the baseline and occurred earlier (time 159.027). This indicates that a balance of influencers, bots, and fact-checkers can help control the spread of misinformation, albeit not as effectively as a high fact-checker configuration.

6. **Mixed Configuration 2:** This configuration had the same peak number of infected users as the high fact-checker setup (404), but the peak occurred slightly later (time 240.192). It shows that even with a high number of bots, a substantial number of fact-checkers can effectively mitigate the spread of fake news.

Based on the simulations and the results obtained, we can draw several conclusions about the spread of fake news in the model:

1. **Influence of Nodes:** The type of nodes in the network significantly impacts the spread of fake news. For instance, increasing the number of influencers in the network (High Influencer Configuration) resulted in a higher peak of infected users, indicating these nodes' influential role in information dissemination.

2. **Role of Bots:** Bots, despite their numbers, were found to be less effective in spreading misinformation when compared to influencers. This may be attributed to their random connections, which may not be as strategically placed or influential as those of influencers. It can also be argued that their constant low trust factor is unjustified, as it is not always apparent whether the user is real or there's a bot hiding behind it.

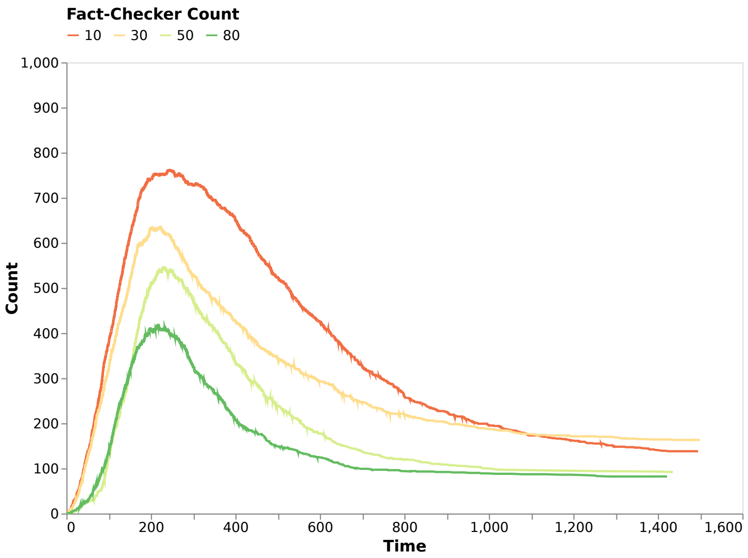


FIGURE 4.3: The impact of initial fact-checkers on the peak of infected users. We increased the number of fact-checkers by reducing the population of regular users. The number of influencers and bots remained the same (10 and 10)

3. **Importance of Fact-Checkers:** The number of affected users rapidly decreased as the number of fact-checkers increased. This demonstrates the value of fact-checking on internet platforms and the requirement for more such organizations to combat false information.

4. **Time of Peak Infection:** The time at which the peak infection occurs also varies depending on the node configuration. For instance, the high bot configuration had a later peak, suggesting that the spread of misinformation by bots is slower but sustained.

5. **Impact of the Confirmation Bias:** Despite the ongoing efforts of fact-checkers, the conversion rate from infected to fact-checker slows and eventually stagnates. It reflects the reality of confirmation bias in information consumption incorporated in the model.

In conclusion, these simulations underscore the complex dynamics of fake news dissemination in online social networks. They demonstrate the need for a multipronged approach to address this issue, including fostering a robust network of fact-checkers, managing the influence of key nodes like influencers and bots, and educating regular users about the potential spread of misinformation.

4.3 Comparison of SIFC Dynamics With the PHEME Dataset

Comparing artificial models against real-world datasets is an essential step in establishing the applicability of the models. In our study, we sought to compare our model's SIFC (Susceptible, Infected, Fact-checker) dynamics with the germanwings-crash-all-rnr-threads event dataset from the PHEME corpus. This event consists of 205 agents, with a peak of 111 infected agents at the end of the observation window. This peak does not show signs of declining, indicating that the misinformation is still in full swing at the time of the last data capture.

It is crucial to note that comparing our model against this dataset presents significant challenges, primarily due to the nature of the data. Firstly, the dataset is limited in size and scope, which may not fully encapsulate the wide range of behaviors and interactions present in a broader or different online social network. Moreover, the roles of the agents within this dataset are not explicitly labeled, making it difficult to directly compare with our model, where agents are categorized as regular users, influencers, bots, or fact-checkers.

Given an ample supply of empirical data, we could validate the model using quantitative metrics such as the Chi-square test, Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). Additionally, comparing the model-derived peak infection times and sizes and the infection and recovery rates with real-world observations would strengthen our confidence in the model's accuracy.

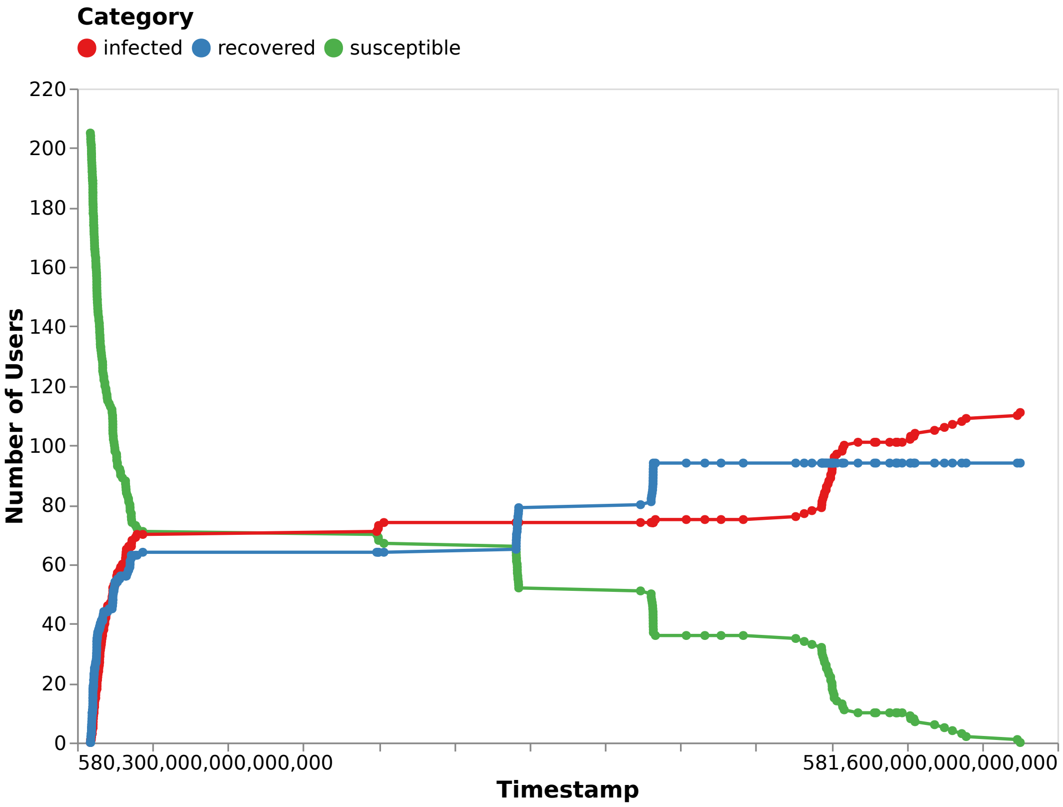


FIGURE 4.4: SIR Dynamics for germanwings-crash-all-rnr-threads

This said, comparing the temporal evolution of the number of agents in each state (S, I, and FC) in both the generated and real data provides some insights. The delayed peak of infected agents in the real data indicates that the spread of misinformation in real-world situations may be slower and more prolonged than that observed in our synthetic model. In our model, the peak of infected agents is followed rapidly by an increase in fact-checkers, which effectively controls the spread of misinformation.

To better align our model with the observed real-world data, we could consider adjusting several model parameters. For instance, we could reduce the rate at which infected agents are converted into fact-checkers or increase the stubbornness of infected agents to fact-checking, both of which would delay the decline of the infected population. Alternatively, we could introduce a delay factor in the fact-checking process, reflecting the reality that fact-checking and debunking efforts often lag behind the initial spread of misinformation.

However, it is important to approach these modifications with caution. While they may make the model output more closely resemble this specific dataset, they may also reduce its generalizability to other situations or datasets.

4.4 Performance Comparison: Python vs C++ Implementation

The need for efficient computation in complex network simulations is critical, especially when the network size or number of iterations increases. In this section, we present a comparative study of the performance of the model written in Python and our C++ implementation.

For this study, the following hardware specifications were used:

1. **Processor:** Intel Core i5 6300U
2. **RAM:** 16.0GB Dual-Channel DDR4 @ 1064MHz (15-15-15-35)
3. **Operating System:** Windows 10

For the comparison, we ran both the Python and the C++ versions with identical configurations and measured the computational time for each. The comparison separately took into account the time of network generation and simulation, excluding data logging.

Since the Python implementation doesn't allow for the configuration of initial fact-checker count, the configurations only include regular users, influencers, and bots.

Configuration	Regular Users	Influencers	Bots
Small Network	500	5	5
Medium Network	1000	10	10
Large Network	2000	20	20

TABLE 4.2: Network Configurations for Performance Comparison

Each configuration was run for 1500 time steps.

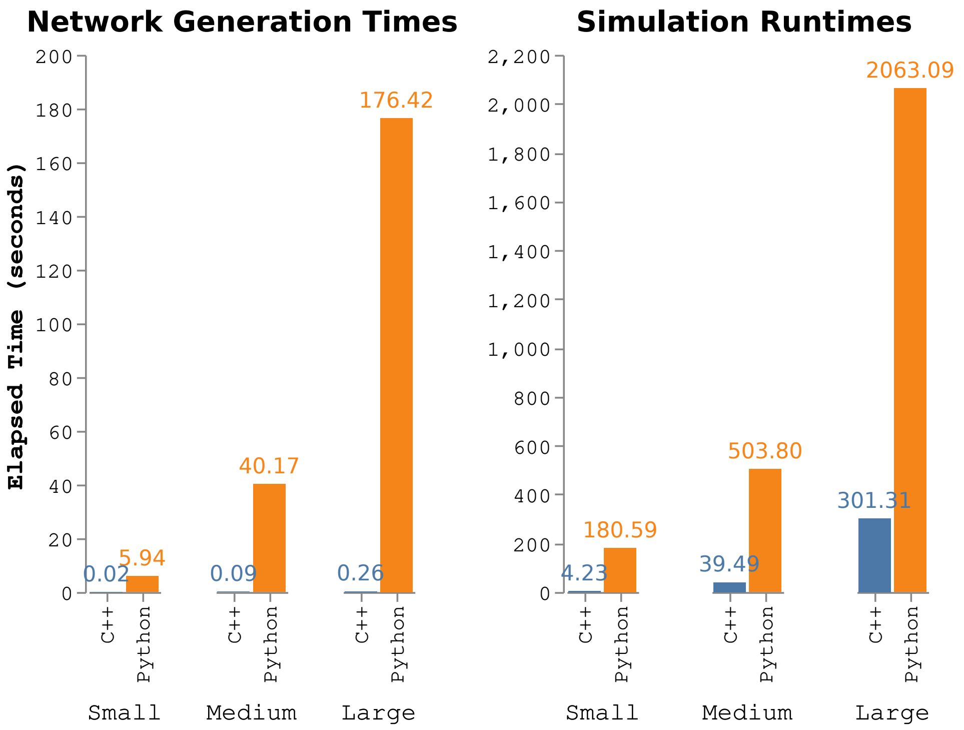


FIGURE 4.5: Comparison of network generation times and simulation runtimes between the C++ and Python versions of the model

Our results show that the C++ implementation significantly outperforms the Python version in computation time. This difference becomes more pronounced as the network size increases, highlighting the suitability of C++ for large-scale network simulations.

However, it's important to note that the efficiency of the C++ model does not only stem from the inherent performance characteristics of the language. The significant performance gain can also be attributed to the algorithmic optimizations.

On the other hand, the Python implementation ran the simulation first, saving a snapshot of the network every n steps, and then visualized the snapshots once the simulation was finished. In contrast, the C++ version updates the GUI instantly when a node emits a signal about a state change, resulting in a responsive but also more resource-intensive system. Because of this, the performance of the C++ version can be improved even further depending on the requirements.

Chapter 5

Discussion

In this research, we have presented a C++ implementation of the referenced model for simulating and studying the spread of fake news in social networks. The C++ version offers significant improvements in terms of efficiency and flexibility, staying faithful to how the model has been described in the paper.

5.1 Conclusions

A key finding from our study is that the structure of the network, specifically the degree distribution of nodes, significantly impacts the propagation of fake news. Our model, while it follows a random network structure, differs from the scale-free nature of real-world social networks like Digg. This discrepancy does not necessarily reduce the usefulness of the model. It brings up an interesting point regarding the variety of real-world networks and the difficulty in creating a model that works for everyone.

Our research on the dynamics of fake news dissemination showed that influencers are crucial to its propagation and that a high number of fact-checkers can significantly lower the peak of infected individuals. These findings highlight potential strategies for combating false news in practical contexts, such as encouraging fact-checking activity and regulating influencers.

The comparison with the PHEME dataset highlighted the difficulties in validating the model due to limited data and unknown roles of agents in the dataset. However, it provided useful insights into potential areas for improvement, such as the need to delay the peak of infected agents in our model. Further work should focus on addressing these limitations to improve the model's predictive power.

5.2 Future Work

While the current model provides a robust foundation for understanding the dynamics of fake news spread, there are several potential avenues for future work that can further enhance its utility and applicability.

Currently, the trust factor is calculated based on the Euclidean distance of spatial proximity/interest attributes between nodes, with the formula 1 - (Euclidean Distance). With the relatively low thresholds used for establishing connections, this calculation often results in a trust factor that is very close to one. While this approach captures the basic intuition that similarity in agents' properties leads to higher trust, it might oversimplify the complex dynamics of trust formation in social networks. Therefore, we can come up with a constant by which the current confidence factor will be multiplied or a completely new approach.

One whole new potential extension is the development of a multi-network model. In the current model, we consider a single network of agents. However, in many real-world scenarios, there may be multiple distinct groups or societies, each with its own network of agents. Take, for instance, the context of the ongoing russian-Ukrainian war, where there is a civilian population and a military front. Communication between these two "boxes" may happen at specific times, such as when news from the frontline is transmitted to the home front. Incorporating such multi-network dynamics into the model could provide a more nuanced understanding of how fake news propagates in these complex scenarios.

Another direction for future work is the incorporation of dynamic connections. In the current model, the network structure is static, with connections established at the beginning of the simulation and remaining fixed throughout. However, in real-world social networks, connections between agents are often dynamic. Our network agents may consider messages from an overly persistent user to be spam and break contact with them.

In addition to these modeling extensions, future work should also address the limitations identified in the current study. This includes efforts to better replicate real-world social networks' scale-free nature and validate the model against more extensive and diverse datasets. Through these efforts, we hope to continue advancing our understanding of the complex dynamics of fake news spread and develop more effective strategies to combat it.

Bibliography

[1] yasinovskyi edu. *Fake News Epidemic Model*. https://github.com/yasinovskyi-edu/fake-news-epidemic-model.

[2] yasinovskyi edu. *PHEME SIR Analysis*. https://github.com/yasinovskyi-edu/pheme-sir-analysis.

[3] P. Erdös and A. Rényi. "On Random Graphs I". In: *Publicationes Mathematicae Debrecen* 6 (1959), p. 290.

[4] Tad Hogg and Kristina Lerman. "Social dynamics of Digg". In: *EPJ Data Science* 1.1 (June 2012). DOI: 10.1140/epjds5. URL: https://doi.org/10.1140/epjds5.

[5] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. *PHEME dataset for Rumour Detection and Veracity Classification*. 2018. DOI: 10.6084/M9.FIGSHARE.6392078.V1. URL: https://figshare.com/articles/PHEME_dataset_for_Rumour_Detection_and_Veracity_Classification/6392078/1.

[6] Quintino Francesco Lotito, Davide Zanella, and Paolo Casari. "Realistic Aspects of Simulation Models for Fake News Epidemics over Social Networks". In: *Future Internet* 13.3 (Mar. 2021), p. 76. DOI: 10.3390/fi13030076. URL: https://doi.org/10.3390/fi13030076.

[7] Simone Raponi et al. "Fake News Propagation: A Review of Epidemic Models, Datasets, and Insights". In: *ACM Transactions on the Web* 16.3 (Aug. 2022), pp. 1–34. DOI: 10.1145/3522756. URL: https://doi.org/10.1145/3522756.