Simulation and Modeling of Information Dissemination in Online Social Networks

---

A

Dissertation

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment

of the requirements for the degree

Doctor of Philosophy

by

Yichen Jiang

December 2022

APPROVAL SHEET

This

Dissertation

is submitted in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

Author: Yichen Jiang

This Dissertation has been read and approved by the examing committee:

Advisor: Michael Porter

Advisor:

Committee Member: Afsaneh Doryab

Committee Member: Laura Barnes

Committee Member: Tariq Iqbal

Committee Member: Heman Shakeri

Committee Member:

Committee Member:

Accepted for the School of Engineering and Applied Science:

Jennifer L. West, School of Engineering and Applied Science

December 2022

Simulation and Modeling of Information Dissemination in Online Social Networks

Yichen Jiang

ABSTRACT

With the development of internet technology, the emergence of social media and online platforms promotes the interchange of information between online users at a fast speed. This allows important and useful information to spread through communities quickly; however, it also permits harmful information, like fake news, to also quickly propagate. Users' responses and attitudes towards the information and subsequent responses may influence how other users perceive and further disseminate the information.

This research aims at investigating the influence that users' networks and stances towards an issue (e.g. fake news, restaurant quality) have on information dissemination in Online Social Networks (OSNs). This research has specifically considered: 1) Discovering the influence that particular user reviews have on future ratings for restaurants on Yelp; 2) Understanding the spread of fake news on Twitter through simulation and modeling. Multivariate Hawkes Processes, a mutually-exciting class of point process models, are used to model the intensity of the information propagation based on measurable features on the network, user stances, and message content. This research models the information dissemination process on social media, quantifies the influence the users received from the user networks, identifies the influential factors, and provides insights into the behavioral patterns between online users during the process.

Dedication

I dedicate my dissertation work to my family and friends. Special gratitude to my mom for all of her support along the way.

Acknowledgments

I would like to take advantage of this opportunity to extend my thanks to all those who helped me during my Ph.D. study and research. First and foremost, I would like to express the deepest appreciation to my advisor, Dr. Michael Porter, for his invaluable guidance, constructive advise, continuous encouragement, and patience that cannot be underestimated which carried me through all the stages of my Ph.D. study. His immense knowledge and insights in this field have made this an inspiring experience for me, and it would be impossible for me to complete my study without his support.

I would like to express gratitude to my committee members - Dr. Afsaneh Doryab, Dr. Heman Shakeri, and Dr. Tariq Iqbal, for the time and effort taken from their busy schedules to my dissertation. Special thanks to Dr. Laura Barnes for her treasured advice and help during the first two years of my Ph.D. life.

Thanks go to my friends here and overseas. Particularly, a big thank goes to Xiang Guo, for his endless amount of encouragement and support to complete my academic journey, and to Tianshu Li and Shiran Ren, for the time and suggestions they have given me.

Finally, I would like to express my appreciation to my parents, for their love and support throughout my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Continuous development and innovation of Internet technology has made it convenient for many people worldwide to seek information online. Existing and emerging Internet technologies are gradually replacing traditional media such as television and newspapers. The internet allows users the flexibility to consume information at their preferred times and locations. Large amounts of information, including text, images, audio, and video, can be absorbed in short periods of time. This on-demand access to information can greatly enhance users' experience and enriches their leisure time. In addition to the one-way consumption of information, online virtual social systems, or social media, can provide users a means to both propagate information and enjoy social experiences without having to be in a particular spatial location. This has greatly reduced the social distance between online users and facilitated the unprecedented flow of information around the world.

Internet technology is a double-edged sword bringing negative effects along with numerous benefits. Due to the ease of access and relative lack of supervision, the Internet can be a platform to disseminate false information. Online users must struggle to make judgements on the veracity of the information they receive from the Internet. Aside from being personally influenced by false information, users may unknowingly spread false information in their online networks. Malicious actors will exploit the

ability to spread false information on online platforms to benefit themselves and their supporters. Both the initiators of false information and their accomplices are often hidden, covered up by the wide dissemination and submerged in the rapid information flow. As innocent users become misled, they will continue to spread the false information which will mislead other users and "pollute" the platform. False information spread online can cause great harm to societies; even leading to violence and abuse.

There exist different forms of false information on social media, but generally, they can be categorized into two classical types which are determined by the spreading platforms: fake news and fake reviews. The former, fake news events, are initiated on blogs or fake news websites, and are brought into and disseminated on social media such as Twitter and Facebook through online users. The false information delivered from fake news will be disseminated by online users and shared with more users at an exponential growth rate, which may cause negative effects. Fake reviews are spread on review websites such as Yelp, Amazon, and TripAdvisor, which can influence users to make decisions on corresponding products or services. Through surfing reviews and comments on review websites, users will obtain a first impression of the product or service they are interested in, which may determine whether to proceed to the next step or not. Businesses and merchants may even employ bots and paid reviewers to post fake reviews to mislead customers in order to obtain more profits. These two classical types of false information are the research objectives of the current study, which will be derived and discussed further.

The information dissemination process has been modeled on different applications such as viral marketing, outbreak detection, finding key blog posts to read in order to

catch important stories, finding leaders or trendsetters, and information feed ranking [W. Chen, Lakshmanan, and Castillo 2013]. Therefore, this promotes us to understand the dissemination progress of false information, how users are affected by the false information, and what consequences will be caused, in order to help prevent the false information from spreading.

1.1 Fake News Dissemination on Twitter

Fake news has never become a novel topic, especially for the online platform in the era of the internet. Fake news, rumors, and information hoaxes are different types of false information, where researchers may provide various definitions for these terms. Although researchers hold different definitions of this term, all of them are in a general resemblance with minor differences. A general definition of fake news, according to a recent research [Allcott and Gentzkow 2017], is: "News articles that are intentionally and verifiably false, and could mislead readers". In the pre-internet era, the consequences of fake news spreading were profound and far-reaching already, and such consequences became even more serious with the development of various means of information dissemination [Burkhardt 2017]. Nowadays, social media and online platforms have become the main source of information dissemination [Tandoc Jr, Lim, and Ling 2018]. The varying extent of misinformation will lead to consequences in varying degrees: people who receive inaccurate information will become uncertain about the validity of the knowledge they should be confident with [Rapp and Salovich 2018]. For instance, fake news is always closely related to Politics: after the 2016 presidential election, there is still a large portion of people who believe that

Clinton's Pizzagate Scandal associated with the child-sex ring was 'probably' or 'definitely' true [Tsfati et al. 2020]; Moreover, fake news has led to the incitement of violence in Nigeria and Nepal [Network 2016]. Prior studies have focused on rumors detection on social media [K. Zhou et al. 2019], fake news detection on Twitter and Facebook using classification methods [Helmstetter and Paulheim 2018, Y. Liu and Y.-F. Wu 2018, Granik and Mesyura 2017], and influence of fake news [Bovet and Makse 2019]. In addition, the dissemination of false information occurring on review websites such as Yelp and Amazon will result in fake/unrecommended reviews, which influences platform users before purchasing products or services. Studies in this domain include fake review detection [Sihombing and Fong 2019, Lu et al. 2013], investigation on review helpfulness [Ngo-Ye and Sinha 2014]. Other relevant studies include fake account detection [Boshmaf et al. 2015] and detection of social bots [Shao, Ciampaglia, et al. 2018]. However, researchers have hardly focused on the fake news dissemination itself, in which the dissemination mode and mechanism are yet to be discovered. This motivates us to explore the possible dissemination patterns of fake news.

The emergence of social media and online platforms promotes the interchange of information from and between online users, which brings the convenience of information dissemination (due to the limited length of the content) with a fast speed to the process [Huang et al. 2017]. This results in a wide range of information dissemination within a short period, which provides sufficient possibilities for the spread of fake news. One critical type of fake news dissemination occurs through fake news websites. Generally, a piece of fake news is generated by fake news websites or blogs and shared to social media platforms such as Twitter and Facebook by website users, and spread by platform users for a certain period. A piece of fake news on micro-blogging

websites was spread within hours which leads to mass panic and confusion [Islam, Muthiah, and Ramakrishnan 2019].

One study on consequences of fake news dissemination indicates that nowadays people are routinely exposed to inaccurate information consciously or unconsciously, and people will usually rely on the inaccurate information they received, which makes them confused, self-doubt about their prior knowledge, convinces people, and influences their responses and subsequent tasks [Rapp and Salovich 2018]. To intervene and combat the process of fake news dissemination, fact-check websites are organized to correct the misinformation/disinformation by posting verified results of suspected fake news. Even though fact-check websites generally react within several days to the occurrence of a piece of fake news, they yield limited effect on such progress. A prior study claims that retractions of fake news often fail to eliminate the negative influence thoroughly [Lewandowsky et al. 2012]; a recent study confirms this finding suggesting that misinformation/disinformation continues to influence people's opinion, concurrently occurring with the dissemination of correction, which indicates that such correction cannot entirely revert public opinion to its original status [Walter and Tukachinsky 2020]. Even worse, false information will be repeated while correcting or retracting it, which enhances the public's familiarity with the false information, thus making the correction process counterproductive [Tsfati et al. 2020].

People's attitudes and responses towards fake news dissemination determine its consequences, which is conducive for us to understand the dissemination process of fake news and its mechanism as well. Thus, how people react to fake news caught our attention. In general, the dissemination of a fake news article on social media is accompanied by the occurrence of conversations between online users, in which the claims from users will reveal and represent their attitudes toward the fake news, which

can be defined as user stances. Multiple prior studies have considered user stances as their research objective: classification approach has been developed for user stances of rumor on Twitter using Hawkes Processes and Maximum Likelihood Estimation [Lukasik et al. 2016]; pathogenic user accounts on social media have been studied on their user behavior to analyze the subsequent negative influence [Alvari and Shakarian 2019]. User stances have also been included as a research approach in relevant studies: it has been used as a significant influential factor for rumor detection [Islam, Muthiah, and Ramakrishnan 2019] and veracity prediction of messages [Dungs et al. 2018].

From the perspective of psychology, user stance, reaction, and interaction as cognitive behavior can be studied for understanding the causal relationship between user behaviors or towards the dissemination consequences of misinformation. Based on past studies of human cognitive behavior, the "Confirmation Bias" effect exists in the process of receiving information: people prefer to believe the information/stance that confirms their pre-received knowledge [Lazer et al. 2018]. Another effect named the "Echo Chamber" effect suggests that people who hold similar opinions will form their group [Bruns 2017], which means one tends to believe and follow like-minded people [Shu, S. Wang, and H. Liu 2017], and this is reasonable to be extended to the context of online social networks. Many prior studies have focused on the reaction/response of online users towards the veracity of fake news: A tri-relationship framework has been created based on publisher bias, news stance, and relevant user engagements for fake news detection [Shu, S. Wang, and H. Liu 2017]. A recent study investigated the behavioral patterns of online users reacting to fake news, suggesting that classical user cognitive behavioral patterns cannot explain all the user behavior, but more individual cognitive structures should be considered further [Zimmer et al. 2019].

Under the context of highly-developed social media, online social networks bring more possibilities for spreading fake news, hence, enhancing the complexity of understanding the dissemination process. This challenge derives us to investigate and explore the mechanism of the dissemination process from the perspective of user stance, in which the existing disseminating patterns should be verified, and more possible patterns will be discovered. Therefore, the role of user stances in the process of fake news dissemination will be studied using a temporal model to explore the possible influence patterns between user stances on Twitter.

1.2 Review Influence on Yelp

With the development of technology, users can seek and receive information remotely from online platforms. Review websites such as Yelp, TripAdvisor and Amazon are such platforms for users to browse the reviews from other customers regarding their experiences before purchasing products or services. Without any previous experiences on the target, the only "useful" information the online users can rely on is the user-generated reviews on review websites. Based on the information they received, platform users can determine whether they will proceed to visit a restaurant, book a hotel, or purchase a product online. In other words, the information users can access are important and influential to their decisions, even becoming crucial in some extreme cases such as observing sequential extremely negative reviews which may enhance their bad impression toward the business.

The emergence of review websites reduces the cost of generating and sharing information, which is even more convenient. Based on this, there is full of redundant, unconfirmed information on the internet which requires receivers' judgment on the veracity

of reviews. Prior studies have focused on investigating whether multi-presentation formats and review enjoyment (readers' perceived enjoyment from reviews) affect review helpfulness [Yang et al. 2017]; the Order effect of reviews has been explored for its influence on review helpfulness [S. Zhou and Guo 2017]. The emergence of unconfirmed information also tests the capability of those online platforms of filtering unwanted information as well. Classification approaches have been adopted for detecting fake reviews on Yelp using text mining method [Aono 2019]; Yelp fake review filter was investigated for its working mechanism using linguistic and behavioral features [Mukherjee et al. 2013].

In addition, user-generated content may not proceed in the direction that a business intends: users will generate the review content based on their subjective feelings which may deviate from reality, intentionally or unintentionally; What's more, competitors of a business would receive benefits from posting malicious reviews towards that business, or posting exaggerated favorable comments for themselves. With the development of online forums and social media, an increasing number of customers prefer looking at the reviews and recommendations before visiting the business, such that these untrustworthy contents will mislead the customers to decide not to visit, which will lead to the unreliability of the overall score (averaged star rating) hence affecting the users' impression toward the business. These negative effects will snowball and eventually give rise to reputation and revenue loss to the business.

Through online interaction behaviors, users could be affected by the information received from other users, and such potential influence will be reflected and revealed in their following behaviors, in a short term or long. Moreover, users' opinions can be artificially guided and induced which brings benefits or other advantages to either the platforms or other related profiteers without being noticed by the users. Thus, we

want to discover the existence of such influence from the reviews of Yelp businesses, and explore the influential aspects of prior reviews on subsequent reviews if it does exist. Furthermore, the influence of reviews will take affect on users' behaviors not only reflected in subsequent reviews, but also on businesses' performance: the purpose is to get profits and enhance reputation through employing bots and paid reviewers to post malicious/favorable reviews to induce the customers. Users may make decisions relying on these exaggerated reviews, hence influencing business evaluation. Thus, it is essential to investigate how the reviews would influence future reviews since reviews would influence users' opinions towards the businesses as well as the ratings.

1.3 Summary

These two types of false information, fake news, and fake reviews will be studied on their subsequent influences in terms of different aspects of user behaviors. Specifically, they will be addressed three research topics:

1) Topic 1 focuses on the influence of Yelp reviews regarding relevant review aspects (e.g., user features, text features) on the subsequent reviews.

2) Topic 2 focuses on the simulation of the fake news dissemination process on Twitter using the Multivariate Hawkes Processes model.

3) Topic 3 focuses on the parameter estimation of the Multivariate Hawkes Processes model to understand the process of fake news dissemination on Twitter.

Chapter 2

Discovering Influence of Yelp Reviews Using Hawkes Point Processes

2.1 Abstract

With the development of technology, social media and online forums are becoming popular platforms for people to share opinions and information. A major question is how much influence these have on other users' behavior. In this paper, we focused on Yelp, an online platform for customers to share information about their visiting experiences on restaurants, to explore the possible relationships between past reviews and future reviews of a restaurant through multiple aspects such as star ratings, user features, and sentiment features. By using the lasso regression model with review features processed through Hawkes Process Model and B-Spline basis functions as the modeling of restaurant basic performance, average star ratings, low star ratings and sentiment features of past reviews have been found to have a significant influence on future reviews. Due to the limited dataset, we performed simulation on restaurant reviews using Multinomial Logistic Regression and rebuilt the model. A verification process has been performed eventually using Logistical Regression. The simulation

and the verification results have been found to support the prior findings which indicate that influence between past and future reviews does exist, and can be revealed on multiple aspects.

2.2 Introduction

With the development of technology, users can seek and receive information remotely from online platforms. Generating and sharing information could be even more convenient at a low cost. Based on this, there are full of redundant, unconfirmed information on the internet which requires receivers' own judgement, and tests the capability of those online platforms of filtering the unwanted information as well. Through online interaction behaviors, users could be affected by the receiving information more or less, and such potential influence will be reflected and revealed in their following behaviors, online or offline, in a short term or long. Moreover, users' opinions can be artificially guided and induced in a possible way that brings benefits or other advantages to either the platforms or other related profiteers without being noticed by the users. Therefore, tracking or mining such influence becomes interesting to us.

Among variate online platforms, Yelp brings out our attention which becomes the ideal target of our research. Yelp is the most popular online forum for information sharing between restaurant customers in the US. By posting reviews through users, Yelp provides an interactive platform for customers to display and share pictures and opinions about local businesses, therefore, the potential customers would receive a general impression of a business based on the reviews and determine whether they are still interested and willing to visit or not, which brings the convenience for the customers. Specifically, Yelp users are able to post reviews of a restaurant regard-

ing their experience during the visit, including but not limited to the meal quality, environment, and services; more importantly, star ratings of the restaurant, ranging from 1 to 5, will be given from the customers along with the review which represents their overall feeling during visits. An averaged star rating computed based on all reviews will be displayed on the main page of each restaurant representing the general feeling of the majority of the customers. The average star rating will leave the first impression on the potential customers and thus is critical and influential: a relatively high proportion of reviews with a high star rating makes it more likely to attract customers while a lower star rating will affect in the opposite way. These types of user-generated content may affect customers' opinions toward a business. However, this may not proceed in the direction that the restaurant intends: users may generate the review content based on their subjective feelings which may deviate from reality, intentionally or unintentionally attracting users; In addition, competitors of one business may employ water army to post false content to discredit the business maliciously. With the development of online forums and social media, an increasing number of customers prefer looking at the reviews and recommendations before visiting the business, such that unreliable content may mislead the customers, and affect the users' first impression of the business. These negative effects will snowball and may eventually give rise to revenue loss to the business. Therefore, it is important to explore such dependency among reviews to investigate how a past review will influence subsequent reviews.

Many of the prior studies have already brought this topic to people's attention by investigating the influence of user-generated content on the information receivers, and such influence could be reflected in the enhancement of users' purchase intention, and the benefits the businesses could receive from. One study of Yelp review

persuasiveness has been performed which indicates the higher trustworthiness of positive reviews other than negative reviews and two-sided reviews with respect to user attitude and purchase intentions [Pentina, Bailey, and L. Zhang 2018]. A study investigating the motives of reading and articulating the reviews on Yelp has found that people are likely to perceive benefits from other reviewers' experiences and share their experience with others, and such likelihood is positively related to users' income [Parikh et al. 2014]. Businesses could also benefit from such review platforms, and the reviews generated by the users of the platforms. More specifically, a one-star increase in the star rating of Yelp reviews has been found that lead to a five to nine percent increase in business revenue [Luca 2016].

The impact a Yelp user could receive from restaurant reviews is obvious by intuition, which can be observed through the growing population of users of restaurant review websites. However, the impact a prior review could produce on future reviews was hardly investigated in the past, which makes this study become challenging. One study of investigating the sequential dependencies in Yelp review ratings suggests that both within-reviewer and within-business ratings are influenced by their previous ratings [Vinson, Dale, and Jones 2019], which reveals the dependency/influence and subjectivity of the reviews. However, this is the only study we have found focusing on the review dependency/influence, and such influence between the reviews is believed to cause the unreliability of such information delivery. If such influence does exist, it is imaginable that any information fragment of a review would hold a certain probability of being received by other reviewers and being duplicated by other reviewers in their own reviews generated subsequently. This process will become repetitive and such information fragments will be transmitted through the review chain continuously, such that anything that deviates from reality (in a positive or negative way) would

be reserved in the review chain over time without being intervened, and potentially influence the business benefit.

Therefore, in this paper, we address three research questions regarding the aforementioned review influence:

1) Does such dependency/influence exists between reviews of a business?

2) If so, where does it exists?

3) How to model such impact?

To answer the above three questions, inversely we determined to build a regression model of the review chain first with variables (detailed addressed in section 3) associated with multiple aspects of a review. By modeling the process of the review chain, the variable(s) that shows the statistical significance, if exist, would be considered as the aspects that reveal such influence among reviews, hence, indicating the existence of review influence of a business; if no variable has been discovered to be statistically significant, then one can state that such influence among reviews still awaits to be discovered.

However, based on what we have learned from prior studies, there are still limitations among all existing approaches addressing these questions. For instance, one study mentioned above [Vinson, Dale, and Jones 2019] that focused on exploring the review dependencies has revealed their existence on review star ratings between pairs of nearby reviews. However, many of other aspects that may take effect have been overlooked, which will be addressed in the current study. These aspects can be summarized as research gaps as follow:

1) More influential aspects should be considered

The aforementioned research concentrated on revealing the review influence through exploring the possible relationship between review star ratings, meaning that the review influence discovered exists on review star ratings only. However, from the nature of the dependency of such information delivery, one can intuitively infer that influential aspects from a prior review to its subsequent reviews will not be limited to a unique aspect, but we may assume that many aspects will be potentially affected, such as content, votes received from Yelp users, and user information [Hu, L. Liu, and J. J. Zhang 2008]. Therefore, more factors should be considered, meaning that features associated with corresponding aspects should be extracted for further analysis.

2) Review will be influenced by more than one review

The prior research which explored review influence by analyzing the sequential dependencies among reviews of a business have treated this problem pairwise [Vinson, Dale, and Jones 2019], that is, all the analyses were performed on current review with one of its targeting prior review in pair; However, people will read more than one past review in order to receive the whole picture of the business as much as possible before making the decision of visiting or not. Therefore, the aggregated influence from past reviews to the future reviews should be considered.

3) Review posted in the far past will still take effect

The aforementioned research [Vinson, Dale, and Jones 2019] only take consideration of the prior reviews nearing the current review but ignored the possible influence received from far past since reviews with high votes will be recommended and presented on the main page of a business by default through Yelp's own sorting function. Yelp users will easily read those reviews without subjectively sorting them, hence, be potentially

affected by them. In addition, people can read prior reviews posted in the far past through keyword searching, which highly raises the possibility of these prior reviews being reviewed by users.

4) Avoid the influence from businesses

The impact between reviews can be captured by analyzing the similarity between reviews. One approach is comparing if the current review will behave in the same way given the star rating of a prior review deviated from the average rating of the business byVinson, Dale, and Jones 2019. The average star rating of a business is the baseline that one should keep concentrated on since all the similarities between reviews may come from the review influence, and the baseline, meaning that users will have similar experiences of visiting the same business. Research should be performed on analyzing the 'pure' review influences based on the modeling of the baseline, to try to avoid bias from the background.

In order to answer the research questions, as well as address the above research gaps, we proposed a novel hybrid model which incorporates the Lasso Regression model with Hawkes Point Process, where the Hawkes Point Process has been implemented for capturing and accumulating the potential impact that a prior review may deliver to its subsequent reviews. The review chain of any business presented on Yelp can be considered as a Hawkes Point Process where the potential impact a review would receive from each of its prior reviews will be computed based on the Hawkes Process model and added onto the variable of each review, thus accumulating and delivering the impact through the review chain. All the variables computed by the Hawkes Point Process will be applied in Lasso Regression model to predict the star rating of each review of a business, where the L-1 regularization of the Lasso Regression Model

holds the shrinkage thus selects the significant variables among all for each business. Instead of the constant intercept, we implemented the B-Spline basis functions for a varying intercept which models the basic case of a business and can be considered as the baseline. In order to verify our finding obtained from the Lasso Regression model, we performed the simulation process as well, in which we implemented Multinomial Logistic Regression to generate fake review ratings based on the distribution of true review ratings for each business. In addition, we shuffled the reviews and matched them with the generated review ratings to perform the Lasso Regression with these generated fake reviews, to verify the findings we obtained from the prior Lasso Regression model we built. All the detailed methodology implemented in this paper is presented in section 4.

We organized this paper as follow: We describe the data used and the variables processed in this study in section 3; then we present our purposed model with details in section 4. The modeling results have been presented in section 5. Finally, we conclude our study and discuss the limitation with possible future works in section 6.

2.3 Literature Review

Our study aimed at detecting possible influence existing between reviews of a business. We achieved this through performing star rating prediction of reviews where multiple features will be extracted and contribute to the final results with varying degrees, thus, such contribution differences will expose where review influence locates.

The prediction of review star ratings has become one of the biggest challenges for researchers who are interested in and willing to explore the Yelp platform. Prior studies have adopted various approaches to achieve this goal: Asghar [Asghar 2016]

has treated this problem as a classification problem with five classes (corresponding to a star rating of 1 to 5 for a review) combining four different approaches of text feature extraction: unigrams, bigrams, trigrams, and latent semantic indexing. Moreover, Machine Learning algorithms combined with sentiment analysis have been implemented into the review rating prediction [Xu, X. Wu, and Q. Wang 2015]. The regression model has been implemented in star rating prediction frequently: Lasso Regression and Vector Auto-Regression have been developed for long-term and imminent future popularity/rating predictions of Yelp reviews, with the implementation of text features such as positive and negative unigrams extracted [Kc and Mukherjee 2016]. Aside from the review rating prediction, Fan and Khademi applied Linear Regression, Support Vector Regression and Decision Tree Regression combined with Term Frequency (TF) and Part-of-Speech to perform the star rating of a business. Machine Learning algorithms including Lasso Regression, Random Forest, and several other models have been implemented for discovering the factors that affect the preferences of romantic partners in their choice of businesses, with respect to business characteristics, review ratings, and romantic-relative languages and words used in the reviews [Rahimi, Clio, and Xi 2017], in which Lasso Regression showed its better performance compared to many other approaches.

In order to capture the possible influence between reviews of a business, one study explored cognitive sequential dependencies by comparing the measures of how much the current review is deviating from the mean between different review distances [Vinson, Dale, and Jones 2019]. In the current paper, we implemented Hawkes Point Process Model to capture and aggregate such impact from a prior review to each of its subsequent reviews in the current paper. Hawkes Point Process has been created by A. G. Hawkes [Hawkes 1971] and has been implemented widely in modeling the

occurrence of an event series such as earthquakes [Freed 2005]. Recently, it has been implemented frequently in modeling and predicting the cascade of streaming social media. One of the studies has implemented Hawkes Point Process in predicting the popularity of Twitter cascades with respect to the expected number of future events [Mishra, Rizoiu, and Xie 2016]. Pinto et al. [Pinto, Chahed, and Altman 2015] developed a Hawkes-based information diffusion model for topic trend detection in social networks which takes the user-topic interactions into consideration; a bi-direction relationship between users and items (user-to-item and item-to-user) was considered and introduced combining with temporal point process model for investigating the latent features beneath the networks [Yichen Wang et al. 2016]. Moreover, Multivariate Hawkes Point Process Model has been implemented in investigating Yelp reviews: it has been developed for capturing the effect of review star rating from users to the star rating of subsequent reviews of a business [Porter 2017].

To the best of our knowledge, all relevant prior studies have focused on investigating the review influence/dependency through analyzing the review star ratings only, however, multiple aspects of future reviews would possibly be affected and present such effect through information sharing; Moreover, any future review will possibly be affected by multiple prior reviews, which is accorded with users' browsing habits, however, this has not been addressed in some of the prior research. To address these gaps, we purposed a hybrid model in which Lasso Regression has been adopted for review rating prediction along with features processed through Hawkes Point Process Model as the independent variables. Given all the great prior studies, text and sentiment features extracted from Yelp reviews, and user features of reviewers have been applied to this study.

2.4 Data Description

2.4.1 Raw Data

In order to verify the existence of review influence/dependency among business reviews, one must explore the possible aspects that review influence may locate. Past research that investigated the review influence among business reviews has revealed the existence of such review influence in review star ratings; Motivated by this, one may reasonably deduce that not only the star ratings, but other aspects could also reflect such review influence. Hence, it is necessary to expand the exploration scope to perform considerable investigation.

Specifically, business data (e.g. business star rating, business review count), user data (e.g. user review count, user fan count), and review data (e.g. review star rating, review text) have been studied and investigated further for building the model. To guarantee that the data targets our research goal accurately, we narrowed down the scope of qualified businesses by two criteria: 1) Businesses with over 500 reviews; 2) Businesses with over 100 reviews posted per year. Thriving businesses with good benefits can easily attract numerous customers to visit and post reviews, thus, providing us with abundant corpus data. The dataset filtered through such criteria guarantees the relatively complete causal chain of review influence among reviews if exists. All the filtered businesses have been matched with their corresponding reviews and user information.

2.4.2 Features

Based on the aforementioned data, data pre-processing has been performed to standardize the data into desired formats suiting for further modeling and application. In this study, we applied four types of features: star rating features, user features, text features, and interaction between features.

Star Rating Features

For star rating features, we converted the star rating of each review using one-hot encoding algorithm. For instance, the star rating of a 4-star review can be converted into:

- 1-star rating: 0;
- 2-star rating: 0;
- 3-star rating: 0;
- 4-star rating: 1;
- 5-star rating: 0.

Therefore, we obtained five different features for each review. In addition, average star rating has been calculated as a feature that indicates the average star rating of all past reviews until the current review, according to the sequential-ordered reviews.

User Features

For User features, we collected review_count, yelping_since, number of fans (since the number of fans is highly correlated with the number of friends, we chose the

number of fans instead of two), and voting count of the aggregation of three categories: useful, cool and funny, of the poster of the current review. Particularly, yelping_since measures the time length between the registration time of the user account and the posting time of the current review. This time length has been converted into numeric consisting by an integer of the number of days from registration day to the posting date of the review, and a fraction of hours and minutes of the posting time.

Text Features

In order to extract information thoroughly from the raw data and obtain a better performance on the model, we considered extracting text features from Yelp reviews. Initially, there are six features considered to be extracted from the review texts, however, due to the high Pearson correlation between some of the features, we considered three features being included in our model: average Word Probability, polarity, and subjectivity.

1) Average Word Probability

Average Word Probability was obtained based on two components: Term Frequency (TF) and Total Term Frequency (TTF). In this case, we defined a review as a document, and all reviews of a business as the corpus instead of reviews of all businesses, since different restaurant-related terms will be mentioned in different businesses. The necessary processing procedures have been performed for calculating TF and TTF, including document tokenization, stemming, and stopword removal. Furthermore, Document Frequency (DF) of each term has been calculated, representing the frequency of documents that contain each of the specific terms. A controlled vocabulary has been constructed based on a filtered list of terms based on setting reasonable

thresholds of DF, hence maintaining the important text information and reducing the processing cost of the model. Unigram features were extracted in this study. Average Word Probability of each review, then, was obtained through the following equation:

$$\begin{aligned} \text{AverageWordProbability}(d) &= \text{mean}(\text{Prob}(t,d)) && (2.1) \\ &= \text{mean}(\frac{\text{TF}(t,d)}{\text{TTF}(t)}) && (2.2) \end{aligned}$$

Where TF(t,d) refers to the Term Frequency of a term within a document, TTF(t) refers to the Total Term Frequency of a term among all the documents of a business, and Prob(t,d) refers to the probability measurement of a term. In this study, all terms in the controlled vocabulary were taken into the above calculation, meaning that terms not appearing in the current document will obtain a 0 score on Prob(t,d); Moreover, the averaging process of each document was calculated based on all terms' Prob(t,d).

2) Sentiment feature: Polarity

Polarity is a sentiment feature that measures the sentiment polarity of a corpus, which ranges from -1.0 to 1.0, corresponding to extreme negative sentiment to extreme positive sentiment. This feature is extracted through a python package called "TextBlob" [Textblob 2021] based on the raw review text content (without any text processing).

3) Sentiment features: Subjectivity

Another sentiment feature is subjectivity, which measures the subjectivity of a corpus, ranging from 0 to 1.0, corresponding to very objective to vert subjective. This feature is extracted through the Python package "TextBlob" as well.

Interactions between Features

Considering the influence of star ratings and the sentiment features, we implement interactions between star ratings and sentiment features, including: 1) Interaction between star rating and Polarity; 2) Interaction between star rating and Subjectivity; 3) Interaction between star rating and Average Word Probability. Interactions are calculated as the multiplier of star rating and the value of the corresponding sentiment feature.

2.4.3 Variables

Based on the aforementioned extraction process, 17 features have been extracted from the raw data regarding various aspects of a review. In order to incorporate the features with Hawkes Point Process Model, all the features have been implemented into the Hawkes Point Process Model to aggregate all the possible impacts received from prior reviews. To further model the decaying speed of impact of different features, five different values of decay parameter have been selected to incorporate with the features, such that a total number of $17 \times 5 = 85$ variables (also called Hawkes features) have been created, in which each feature will be fitted with five decay values. This will be further discussed in the next section.

2.5 Methodology

Discovering and Understanding the inner relationship between business reviews regarding various review aspects is a complicated but meaningful problem. This paper aims at building an appropriate model to reveal the existence of such review influence

among business reviews, and investigate how reviews would affect one another, with respect to multiple influential aspects. Based on the prior studies that addressed these research questions, it is not hard to find out the limitations and gaps in the methodology they applied for investigating the problems: 1) Reviews would be influenced by not only a single aspect, but more aspects will be influenced and revealed such influence; 2) Any of the reviews would possibly be influenced by not only any other single review, but all prior reviews together; 3) Reviews posted from far past would still take effect on current review; 4) Review influence should be investigated under the case that similarities derived from the background (business) have been avoided.

All the aforementioned aspects are the targets addressed in the proposed method, which will be reflected in the modeling part, and be presented next.

2.5.1 Lasso Regression Model

We implemented Lasso Regression to calculate and optimize the coefficient for each variable as the event magnitude α_j of each variable, in which Lasso Regression is a special type of Linear Regression with L-1 regularization as shrinkage. The objective of the regression is to minimize the Sum of Squares by constraints:

$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{i=1}^{p} |\beta_j| \tag{2.3}$$

where λ is the tuning parameter that controls the L-1 penalty, and generally the smallest λ will be chosen.

2.5.2 Lasso Regression Model with Hawkes Features (Variables)

To achieve the goal of incorporating all the aforementioned problems, we proposed a novel method that integrates the Lasso Regression Model with Hawkes Point Process Model [Hawkes 1971]. The entire model implements the features extracted from the raw data and processed by applying Hawkes Point Process Model to capture the influence received from all the prior reviews (including those posted in the far past), and performs the prediction of review star ratings through the Lasso Regression Model. The model format is presented as follows:

$$y_i = \sum \beta x_t + C \tag{2.4}$$

$$= \sum \beta \left[\lambda(t)\right] + C \tag{2.5}$$

where β represents the coefficients of variables, $\lambda(t)$ represents the variables (Hawkes features) we extracted from the yelp dataset. The Hawkes Process Model applied for extracting features can be expanded as follow:

$$\lambda(t) = b(t) + \sum_{i:t>t_i} \alpha g(t - t_i) \tag{2.6}$$

$$= \sum_{i:t>t_i} \alpha g(t - t_i) \tag{2.7}$$

$$= \sum_{i:t>t_i} \alpha \delta e^{-\delta(t-t_i)} \tag{2.8}$$

Since a review series of a business only contains one immigrant event which is the first review of that business, it can be modeled by Marked Hawkes Process where

the Hawkes intensity function only contains its summation term; α is the branching factor of the process which controls the number of events the process may generate; $g(t - t_i)$ is the kernel function, and we applied exponential distribution as the kernel function to model the information diffusion process of the Yelp reviews, with decay parameter δ. Then, the Lasso Regression Model can be expressed as follow:

$$y_i = \sum_j \sum_k \beta_{jk} \left[\sum_{i:t>t_i} \alpha_j \delta_k e^{-\delta_k (t-t_i)} \right] + \sum \theta_q B_{qi} \tag{2.9}$$

The structure of the above model is basically built on Lasso Regression Model, where β is the coefficient obtained from Lasso Regression Model for each variable; The term locates within the brackets $\sum_{i:t>t_i} \alpha_j \delta_k e^{-\delta_k (t-t_i)}$ is the variables applying in the prediction of review star ratings, referring to the Hawkes features, which are the features extracted from the raw data and processed through Hawkes Point Process Model. The summation term $\sum \theta_q B_q$ is the intercept term of the Lasso Regression Model, which is substituted and modeled by B-Spline basis functions as the baseline of the business, where q represents the knot of the B-Spline functions. Particularly, we set year-break of review posting times as the knots of the B-Spline functions, and the spline order has been set to 3. Moreover, the review posting times are set to be the control points of the B-Spline functions, which were accurate to seconds to differentiate reviews posted on the same day. More information of the B-Spline basis function is presented explicitly in Appendix.

Specifically, α_j is the branching factor of the j-th feature. Particularly, α_j has been set to $\alpha_j = 1$ for the star rating of a review, or $\alpha_j =$ feature value for all the other features of a review (e.g. review count, sentiment score etc.). Term $\delta e^{-\delta(t-t_i)}$ is the function of the exponential kernel of the Hawkes Point Process Model, which

determines the decaying process of an information diffusion process. Term t_i is the occurrence time of i-th event, in our case, the posting time of i-th review of a business. Term δ_k is the k-th value of the decay parameter δ; In our case, five values have been selected for the decay parameter: $\delta \in [0.005, 0.05, 0.1, 1, 5]$, representing five different decay speed of the impacts received from prior reviews. Based on the aforementioned review aspects, we defined and extracted three types of raw features from the Yelp review dataset: star rating features, user features, and text features, 17 features in total. Each feature was matched with five different values of decay parameter when calculating the aggregated influence received from past reviews through Hawkes Point Process Model, hence, a total number of $17 \times 5 = 85$ variables have been created and implemented into the Lasso Regression Model. Based on the shrinkage property of the Lasso Regression Model, variables with the most appropriate value of the decay parameter (which fits the real decaying speed) will be selected as a significant feature.

2.5.3 Simulation Using Multinomial Logistic Regression Model

Considering that we only have a small certain number of businesses with over 500 reviews in total, we decided to perform the simulation based on the Multinomial Logistic Regression Model, which would remedy the situation of limited data and present a general view of the businesses as well.

During the simulation, we implemented the Multinomial Logistic Regression Model with the star ratings as the dependent variable and the B-Spline basis elements alone as the independent variables. Multinomial Logistic Regression is the extension of general Logistic Regression for predicting categorical variables with multiple categories instead. The predicted probability of receiving a k star rating ($k \in [1, 5]$) of a review

can be expressed as follow:

$$P(Y_i = k) = \frac{e^{\beta_k \cdot X_i}}{\sum_{j=1}^{K} e^{\beta_j \cdot X_i}} \tag{2.10}$$

where

$$\begin{aligned} \beta_k \cdot X_i &= \beta_0^{(k)} + \beta_1^{(k)} x_1 + \beta_2^{(k)} x_2 + \cdots + \beta_p^{(k)} x_p && (2.11) \\ &= \beta_0^{(k)} + \sum_{i=1}^{p} \beta_i^{(k)} x_i && (2.12) \\ &= \beta_0^{(k)} + \sum_{i=1}^{p} \beta_i^{(k)} (\theta_i B_i) && (2.13) \end{aligned}$$

where i denotes the variable number ranging from 0 to a total number of p; j denotes the star ratings ranging from 1 to a total number of $K = 5$.This model was implemented for simulation in the current paper, which returned the probabilities of receiving the five different star ratings for each review in the business. Thus, we can generate a new star rating for each review based on the cumulative probabilities computed from the results. A total number of 100 times of simulations were performed using Multinomial Logistic Regression to generate fake (simulated) star ratings. As the number of simulations was going up for each business, we were obtaining a set of generated star ratings for each review, (e.g. 100 star ratings will be returned for each review if 100 runs of simulation have been conducted) and the resulting star ratings among all the reviews of a business follow exactly the distribution of the cumulative probabilities (obtained from the result of Multinomial Logistic Regression) when the number of simulation is sufficient enough (the star ratings were generated based on it).

2.5.4 Lasso Regression Modeling on Simulated Data

After the simulation procedure, we performed Lasso Regression Modeling on simulated data. Since the simulation process for each business was performed 100 times which returned 100×number of reviews of simulated star ratings in a business, the Lasso Regression model was built on simulated data for each run of the simulation with simulated star ratings as dependent variables. Particularly, for each run of the simulation, business reviews were shuffled and re-positioned to each order, matched with the simulated star rating in the same order, and inputted into the model as independent variables.

2.6 Results

This section presents the modeling and simulation results from multiple steps. The general procedures we performed in this study include:

1) Multivariate Hawkes Process and Lasso Regression Modeling on processed Yelp data;

2) Generating simulated data through Multinomial Logistic Regression;

3) Multivariate Hawkes Process and Lasso Regression Modeling on simulated Yelp data;

4) Verification through Logistic Regression.

We will introduce the above procedures in details and discuss the findings obtained from the results.

2.6.1 Lasso Regression Modeling on processed Yelp data with Hawkes features (Variables)

The hybrid model built based on Multivariate Hawkes Point Process and Lasso Regression Model has been implemented on the Yelp review data through Python and R. All the Yelp data has been collected through the public Yelp Dataset Challenge [Yelp 2020] of the year 2019 and 2020 from which a total number of 1715 unique businesses containing more than 500 reviews have been extracted and analyzed, being matched with corresponding review and user information; text features and interactions between features have been extracted and computed as well. All the raw features have been processed through Multivariate Hawkes Point Process to acquire the variables which aggregate the influence from prior reviews.

With the pre-process through Multivariate Hawkes Point Process and the B-Spline basis functions, we could input all the data for building a Lasso Regression Model to make predictions and check the significance of each variable where the significant variables will indicate the influence gained from prior reviews. The results have been summarized in Table 2.1 and Fig. 2.1.

Lasso Regression Modeling allows selecting a set of the most effective variables among all similar variables such that reducing the complexity and the likelihood of being overfitted. Table 2.1 provides the general view of the significance of each variable: The count of each variable represents the number of businesses that obtained a significant result on that variable among all businesses regardless of the decay parameter of that variable, and the proportion was computed through dividing the count by 1715, which is the total number of businesses in our dataset. From the proportion, one can find that the average star rating (with the proportion of 0.1569) has the most significant

Table 2.1: Variable Significance of Original Dataset

Variable	Count	Proportion
1 Star	211	0.1230
2 Star	128	0.0746
3 Star	123	0.0717
4 Star	94	0.0548
5 Star	106	0.0618
Average Star Ratings	269	0.1569
Votes	96	0.0560
Elite Count	53	0.0309
Fan Count	56	0.0327
Review Count	70	0.0408
Yelping_Since	53	0.0309
Average Word Probability	66	0.0385
Sentiment: Polarity	91	0.0531
Sentiment: Subjectivity	128	0.0746
Stars × Average Word Probability	49	0.0286
Stars × Polarity	79	0.0461
Stars × Subjectivity	117	0.0682

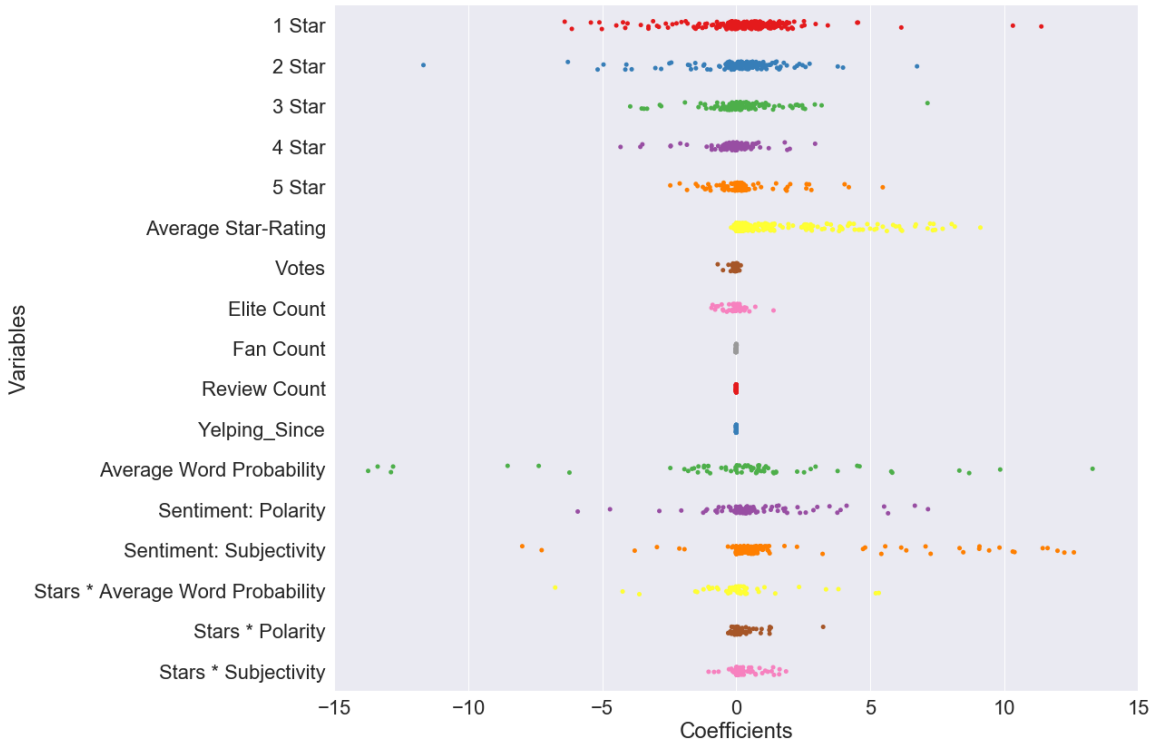

Figure 2.1: Coefficient Result of Lasso Regression Model Built on Original Dataset

impact on future reviews, followed by the 1-star review (proportion = 0.1230), 2-star review (proportion = 0.0746), and the 3-star review (proportion = 0.0717), which indicates that reviews with a lower star rating are more likely to prompt Yelp users who saw the reviews to post new reviews; sentiment-related variables such as sentiment_subjectivity (proportion = 0.0746), sentiment_polarity (proportion = 0.0531) and the interaction between sentiment_subjectivity and star ratings would trigger new reviews as well since reviews with strong personal feelings or along with extreme star ratings will be more infectious.

The dot plot provides a view of the variables toward influence direction. The dots presented in Fig. 2.1 represent the non-zero coefficients obtained from the result of the Lasso Regression Model in which coefficients of each variable have been plotted together regardless of the decay parameter, and the boundary of the X-axis has been

set to [-15,15] to eliminate outliers for visualization purpose. A positive coefficient indicates a positive influence of prior reviews applied on future reviews, while a negative coefficient is performed in the opposite way. Therefore, we can observe that the majority of variables could not guarantee a certain influence direction on future reviews; However, average star ratings are highly likely to have a positive influence on future reviews, which reveals the nature that a business with a higher rating will keep attracting customers to visit and post positive reviews toward the business on Yelp, and thus helps improve rating or at least remain it unchanged; sentiment_subjectivity and sentiment_polarity have similar trends but less obvious as averaged star rating does, which indicate that reviews with subjective sentiment or positive sentiment are more likely to exert a positive influence on future reviews, in other words, reviews with high star rating would be triggered.

All findings obtained from the basic Lasso Regression Model indicate the existence of inner relationships between prior reviews and future reviews with respect to different review aspects (variables), which motivated us to perform further analysis.

2.6.2 Generate simulated data through Multinomial Logistic Regression

Due to the limited number of qualified businesses we have (1715 businesses in total), we performed the simulation based on the Multinomial Logistic Regression Model for the businesses with significant yelp variable(s) in the result of the Lasso Regression Model to model the basic standard of the businesses, which helps recognize and differentiate the influence from the reviews themselves and the businesses, and this would also remedy the situation of a small dataset and present a general view of the

businesses. Furthermore, we concentrated on businesses with reviews posted in high frequency to detect more accurate inner relationships between reviews, such that a total number of 152 businesses with at least one significant variable (variable with non-zero coefficient) and over 100 reviews posted per year were selected for simulation, with significant coefficients scattered in Fig. 2.2. It can be obviously observed that the majority of the significant coefficients of selected businesses are positive, which indicates that these businesses are more attractive than others with frequent-posted reviews that will hold a positive influence from prior reviews on future reviews, even for all variables we considered.

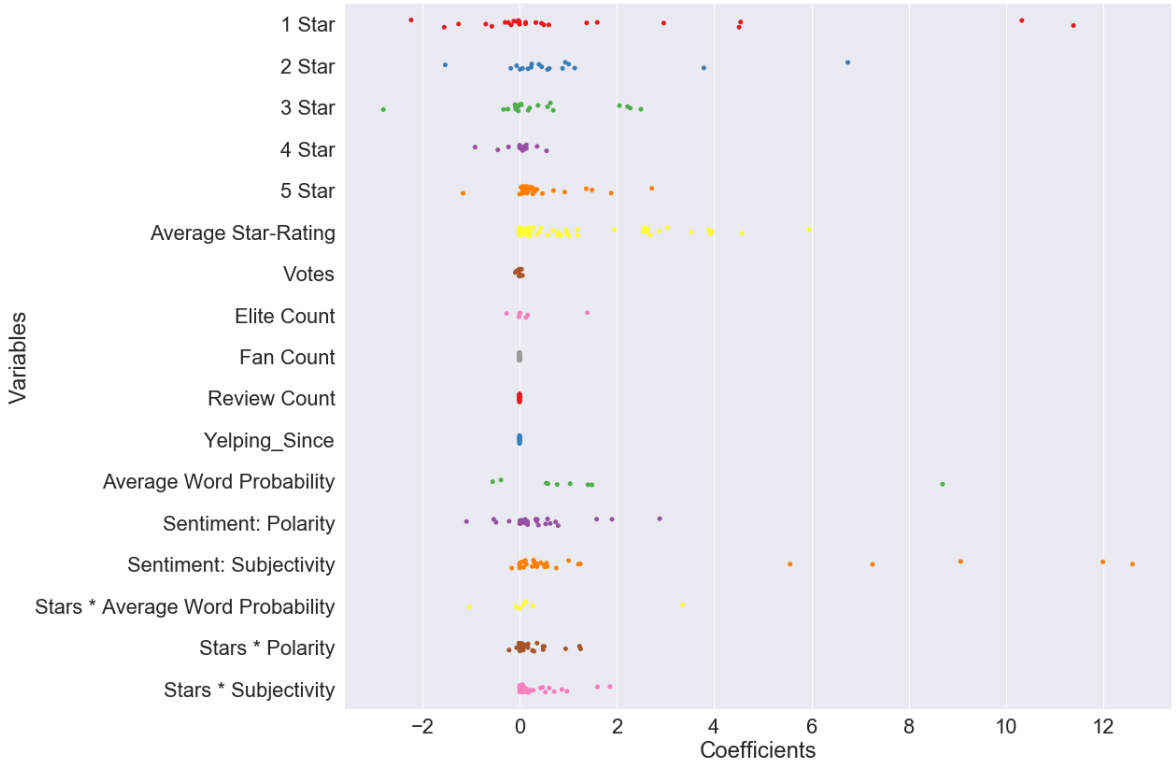

Figure 2.2: Coefficient Result of Selected Businesses

We implemented the Multinomial Logistic Regression Model for simulation with the star ratings as the dependent variable and the B-Spline basis elements alone as the independent variables, returning the probabilities of receiving five different star rat-

ings for each review in the business, and then generated new star rating for each review based on the cumulative probabilities computed from the simulation results. One hundred times of simulations were performed for each selected business. As the number of simulations went up for each business, we obtained a set of generated star ratings for each review, (e.g. 500 star ratings for each review if 500 times of simulation have been conducted) and the resulting star ratings among all the reviews of a business follow exactly the distribution of the cumulative probabilities when the number of simulation is sufficient enough (we generated the star rating based on that). A total number of 15200 times of simulations have been performed for model re-building in the next step.

2.6.3 Lasso Regression Modeling on simulated Review Star Ratings and Hawkes features (Variables)

In order to obtain the basic standard of the businesses, we re-computed the decay variables for all features using the Multivariate Hawkes Point Process Model, in which the influence of past events (reviews) will be quantified and aggregated on the following events; however, contrary to the earlier implementation, we performed some changes to this model:

1) Differing from the previous processing which was performed only once for each business, we re-computed the decay variables based on the number of simulations we ran for each business; For each time of the simulation, we replaced the true star ratings with the simulated star ratings;

2) We shuffled the reviews such that different reviews were placed on the original order corresponding to the simulated star rating.

Following the above changes, we built the Lasso Regression Model over again for simulated data obtained from simulated star ratings and shuffled Yelp reviews processed through Multivariate Hawkes Point Process Model. The coefficients generated by the rebuilt Lasso Regression Model were compared to the observed coefficients obtained from the original Lasso Regression Model for checking the significance of each variable. Specifically, for each business, we have 100 times of simulations on which the Lasso Regression Model was built over again to compute the coefficients for each variable with different values of decay parameter, hence we have 100 coefficients obtained from re-built Lasso Regression Model on each variable with a unique value of decay parameter; each coefficient was compared to the observed coefficient obtained from the original Lasso Regression Model: the number of absolute values of simulated coefficients obtained from re-built Lasso Regression Model that is larger than or equal to its corresponding observed coefficient will be divided by the total number of simulations so as the p-value of the current variable with a unique value of decay parameter to determine its significance. Since we have five values of the decay parameter, we could obtain five p-values with respect to the different values of the decay parameter, we hence determined the significance of the current variable regardless of the decay parameter by the lowest p-values among all five p-values: if one of the five p-values indicates its significance, we could then conclude the significance of current variable regardless of the significance of other four p-values. The results of simulated coefficients have been plotted in Fig. 2.3.

Fig. 2.3 presents the density of non-zero coefficients of simulated data compared to the corresponding observed coefficients regardless of the decay parameter, with the threshold set to -15 and 15. Due to the fact that the data implemented for generating the simulated star ratings was the B-Spline basis elements which were considered to

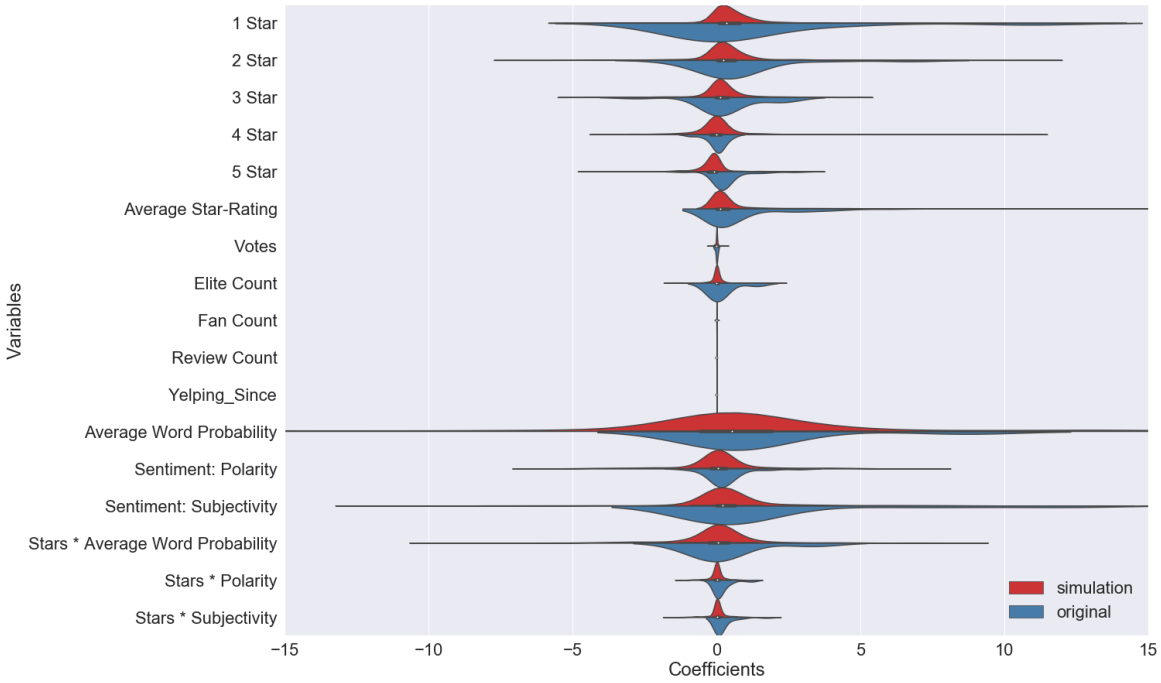

Figure 2.3: Result of Simulated Coefficient for Selected Businesses, Obtained from Re-Built Lasso Regression Model

represent the basic standard of the businesses, the coefficients are scattered around 0 (red region) as what we expected for the business's standard, while the corresponding observed coefficients are expanded with different degree of variations for different variables.

2.6.4 Verification through Logistic Regression Model

We performed verification on the results obtained from the Lasso Regression Model built on the original dataset (with 1715 businesses). We created a binary label for each business where a business received a label of 1 if it has at least one non-zero coefficient on the modeling result of the variables, or a label of 0 with no non-zero coefficients existing within the result, regardless of the variable type, and this label was set to be the dependent variable of the Logistic Regression Model. Business attributes were

extracted as the independent variables of the Logistic Regression Model and were selected based on multiple times of modeling, which include:

- Stars: the average star rating of the business, in which the range of star ratings from 0 to 5 has been divided into levels with increments of 0.5 (e.g. 0, 0.5, 1.0), such that the average star ratings were truncated corresponding to the closest level (e.g. an average star rating of 3.82 will be matched to the level of 4.0);
- Review_count: total number of reviews received from Yelp users;
- Year_count: total number of years the business has been operated, which has been extracted through the posting time of the earliest and latest review of the business;
- Price_range: the dollar sign presented on the home page of the business on Yelp, representing the general cost for visiting it: more dollar signs indicate a possible higher cost;
- State: the state where the business locates;
- Attire: casual or dressy;
- Breakfast, Brunch, Lunch, Dinner, Late-night, Dessert: Boolean variable indicating whether the business provides the corresponding service or not.

The result of Logistic Regression Model has been summarized in Table 2.2: the star rating of the business is significant with coefficient of -0.1969 which indicates it is more likely that past reviews of a business with lower average star rating will influence the futures of current business, furthermore, a lower average star rating is caused by accumulative reviews with low star rating which verify the finding from Table 2.1

Table 2.2: Result of Logistic Regression Model

Variable	Coefficient	P-Value
Stars	-0.1969	0.000
Review_Count	0.0004	0.000
Year_Count	-0.0397	0.002
State	-0.0505	0.176
Attire	0.1021	0.111
Price Range	0.1546	0.035
Dessert	0.0628	0.555
Late night	0.1248	0.264
Lunch	-0.0965	0.397
Dinner	-0.1368	0.353
Brunch	-0.0912	0.453
Breakfast	0.2446	0.102

that 1-star, 2-star and 3-star ratings are the variables that have significant influence on the future reviews; review count is significant which indicates that a business attracts relatively more customers to visit and post reviews will hold influence on the future reviews; year count is significant however with a negative coefficient, from which we could infer that businesses with relatively long-term operation have more reviews posted at the early stage of Yelp's development with less influence due to the limited number of users.

2.7 Discussion

We held the assumption that there is an influence/dependency between past and future reviews before conducting the experiments. In order to verify this, we extracted features through Hawkes Process Model to aggregate the possible influence for each review from all its prior reviews, and built Lasso Regression Model on these Hawkes

features (variables). Based on the results of our proposed method, we proved that the influence between reviews does exist; Furthermore, it can be revealed from multiple aspects of a review, including low star ratings (which was investigated and found by prior research), sentiment scores, as well as interactions between sentiment scores and star ratings. These findings advance the existing methodology, and provide the possibility for further analysis.

The limitation of the current paper can be addressed in several aspects. First, the businesses in the public Yelp dataset were partially selected such that most are located in Las Vegas and Phoenix, which may cause bias on the result and inapplicability of conclusions for businesses elsewhere. Second, filtered businesses came from 2019 and 2020 Yelp datasets which contain inconsistency of year range: businesses in the dataset of the year 2019 lacked the information of the dataset in the year 2020. In addition, the simulation time was set to 100, which can be enhanced to reach a more accurate and reliable result.

Based on the aforementioned limitation, we can improve the research in multiple directions: apply a larger and more general business dataset; increase the number of simulations; particularly, further analysis could be performed to explore specific relationships between prior reviews and future reviews with respect to different aspects (e.g. how a review feature would affect other features specifically).

2.8 Conclusion

In this study, we performed analysis on Yelp data to investigate the influence of prior reviews on future reviews of a restaurant. Review features from multiple aspects were extracted and processed through Hawkes Process Model to aggregate influence

from prior reviews, and were applied to Lasso Regression Model along with B-Spline basis functions as the baseline of the restaurant. The basic results proved that such review influence does exist, and can be found in multiple aspects of a review such as sentiment score and star ratings. These findings have been presented through the simulation as well, and have been partially verified through Logistic Regression Model.

Chapter 3

Simulating Fake News Dissemination on Twitter with Multivariate Hawkes Processes

3.1 Introduction

Online Social Networks (OSNs) have surpassed traditional media outlets to become the main source of information dissemination [Tandoc Jr, Lim, and Ling 2018]. While OSNs help bridges the information gap and speeds up communication between people around the world, they can also nurture the dissemination of misinformation. The varying extent of misinformation will lead to consequences in varying degrees: people who receive inaccurate information will become uncertain about the validity of the knowledge they should be confident with [Rapp and Salovich 2018]. The dissemination of fake news on OSNs has caused negative and even severe effects such as the incitement of violence in Nigeria and Nepal [Network 2016]. There is a growing body of research on fake news dissemination, detection, prediction, and prevention. For example, [Shu, Sliva, et al. 2017] summarizes the typical features for fake news detection on OSNs from a data mining perspective, which includes content features such as linguistic style and visual elements, and social context features including bot ac-

counts, posting behaviors, and dissemination characteristics through users networks following echo chamber effect. In order to quickly react to fake news spreading, studies such as [Murayama et al. 2021, Tian, X. Zhang, and Peng 2020] create real-time predictions on misinformation/disinformation to make prevention possibly from its earliest stage or predict the popularity dynamics of a cascade to estimate the possible long-term consequences caused by it. It is important to note that on OSNs, any user may create or reproduce fake information, and users who follow him/her may be influenced enough to propagate it to other connected people. This makes the user relationships [Davoudi, Moosavi, and Sadreddini 2022] an important aspect when studying the information dissemination process and consequences, especially for misinformation and disinformation in which users' attitudes toward the information are a part of the content that is delivered.

As mentioned above, there are a variety of methods and models that can be considered for modeling of the fake news dissemination process. However, one has to prove the applicability to make an argument for applying the model in the given scenarios, where the model should capture the process of disinformation/misinformation dissemination process under different cases including extreme cases. In this case, model testing simply on a limited real dataset is not reliable enough with uncertainty. Therefore, simulation has been created and developed to fill this gap, which is a risk-free approach for model testing with less uncertainty. Therefore, simulation has been created and developed to fill this gap, which is a risk-free approach for model testing with less uncertainty. Simulation approach allows flexible hyper-parameter tuning to imitate different cases. One typical popular model for such simulation is called the Agent-Based simulation model [X. Li et al. 2008], which captures the information dissemination process among users and considers each user as an agent which fits well

for Twitter-like OSNs where opinion leaders usually take the dominant place in influencing others through information spreading. However, an Agent-Based model may be hard to quantify the "intensity" of the information delivered from an individual: How popular it is? How strong and influential such a piece of information would affect online users? Can the consequences of misinformation/disinformation dissemination on OSNs be predicted and estimated? In this case, statistical modeling involving the temporal features should be able to answer these questions thoroughly. Hawkes Processes model [Hawkes 1971], a time series model which captures the occurrence of event sequence has been implemented frequently in modeling the information dissemination process on twitter-like OSNs. The study in [El Maazouz and Bennouna 2018] improved the thinning method, an approach for simulating events for Hawkes Processes, to simulate rare events on Twitter. The paper [Hagberg, Swart, and S Chult 2008] extended the usage of the Hawkes Processes model which created a new simulation approach incorporating interactions between users who followed each other and applied to wider circumstances with the occurrence of rare events. These are good examples of implementing the Hawkes Processes model with a new-developed simulation approach in the information dissemination process on OSNs, which took the advantage of Hawkes Process in capturing the self-exciting property in the information dissemination process within the user interaction. However, features and aspects that the simulation for the fake news dissemination process on OSNs should incorporate are more than what people observed from the regular information spreading process, and this has not yet been studied and developed comprehensively.

This paper introduces a novel simulation method that combines a Hawkes Point Process model with an agent-based model that captures both the temporal patterns and user networks to produce realistic fake news spreading behavior on OSNs. Particu-

larly, this method will focus on Twitter OSN and include tweet types, user stances towards fake news stories, and user networks, and can be generalized to other OSNs with proper adjustments. Based on this, we believe that this simulation approach will benefit studies or researchers digging into the area of disinformation/ misinformation to simulate different scenarios for model testing purposes; moreover, the reproduction of the fake news dissemination process will also help us understand the user behavior patterns and dissemination mechanism of how information pieces are diffusing between individuals and user groups. The following sections of the paper are organized as follows. Prior studies and research will be reviewed in section 3.2. Necessary definitions and assumptions of our model and simulation approach will be introduced in section 3.3, followed by the specific simulation illustrations and the pseudo code in section 3.4. A simulation example will be presented in section 3.5.3. In section 3.6, we will also discuss the application and the limitation of this simulation approach, and conclude the paper with possible future work.

3.2 Literature Review

This section reviews the simulation methods relevant to information dissemination on Twitter; namely the simulation of Twitter data and the simulation of Hawkes Point Processes.

3.2.1 Simulation of Twitter Data

Agent-based simulation approaches have been used to simulate the actions and interactions between agents within the social network. For example, Serrano and Iglesias

2016 presents an agent-based model to validate viral marketing strategies in Twitter and in [Luke et al. 2005] where a social simulator was employed to study rumor diffusion. Similar studies include [D. Liu and X. Chen 2011] that builds an Agent-based model for rumor spreading on Twitter using SIR model structure (Susceptible, Infected, Recovered), [Weng, Menczer, and Ahn 2013] which implemented four different models considering different magnitude and mechanisms of social networks to predict rumor spreading, [Ahmed and Abhari 2014] that implemented an agent-based simulation to simulate Twitter data regarding user behaviors and evaluate different information retrieval methods to optimize a recommendation system, and [Beskow and Carley 2019] which introduced an Agent-based Model particularly for Twitter social environment called *twitter_sim* to evaluate the emerging consequence under malicious agents such as bots and trolls with respect to user behaviors such as tweets, replies, retweets, mention, following, etc.

Other approaches to simulate Twitter data/system include [Sakas and Sarlis 2016] which adopted a dynamic simulation model to evaluate the performance of library promotion on library services, [Yufang Wang et al. 2020] used a random sampling process to generate Twitter data for real-time prediction of flu epidemics through using geo-tagged Twitter streaming data, and [Sano et al. 2021] which implemented a modified voter model to simulate Twitter data to replicate the scenario of information spreading on Twitter regarding Radiation of Fukushima nuclear power plant accident.

These approaches that use Agent-based models to simulate Twitter data have many attractive properties, but are limited in the temporal patterns they can capture. Specifically, they are limited in their ability to explain how fake news popularity and users' attention changes over time and what consequences it may lead to eventually. This has led us to consider Hawkes point process models to more realistically simulate

the spread of fake news on Twitter.

3.2.2 Simulation of Hawkes Point Processes

Hawkes Point Processes model was developed in [Hawkes 1971] and used to model many types of event series such as earthquakes. It has been frequently implemented in the information dissemination process on OSNs in recent years to investigate the dissemination mechanism. In order to better capture the occurrence of event series and predict the possible results, simulation methods were created and developed to imitate different scenarios. The most popular simulation method was introduced in 1981 by Ogata [Ogata 1981], in which a Hawkes Process is defined as a conditional stochastic intensity-based model [Shlomovich et al. 2022], such that each potential review will be accepted or denied through updating the maximum intensity iteratively, and this method is also called thinning method. The running time of thinning method is relatively high $O(N^2)$ which makes the process inefficient, and the process generates events through both exogenous and endogenous factors such that one cannot differentiate whether an event is triggered by background rate or not [Simon 2016]. To overcome such problems, a cluster-based simulation approach was created [Møller and Rasmussen 2005] which considers a Hawkes process to be constructed on a marked Poisson cluster process [Hawkes and Oakes 1974]. Each immigrant will generate a series of events that form one cluster, and each current generation will generate a Poisson process of off-springs of the next generation. This method considers the branching structure of the Hawkes Processes model, which has been optimized to reduce the running time [Møller and Rasmussen 2006], and extended to the Multivariate Hawkes Processes model [Dassios and H. Zhao 2013]. Both the intensity-based simulation and cluster-based approach have been applied in many prior studies such

as [Zipkin et al. 2016], [Morse 2017, Kirchner 2017]. Particularly, [Zipkin et al. 2016] performed Hawkes Process modeling, simulation, and estimation based on different artificial networks to investigate social network interactions.

In [Kong, Rizoiu, and Xie 2020], a generalized SIR model incorporating Hawkes processes was introduced along with its appropriate simulation algorithm. Many such simulation approaches were developed or revised based on the intensity-based or cluster-based approaches introduced above, such as [Bowsher 2007, Law and Viens 2016, C. Li, Song, and X. Wang 2019], which helped extend the usage of Hawkes Processes models to different areas such as analysis of emergency calls [C. Li, Song, and X. Wang 2019], social networks [Pinto and Chahed 2015, El Maazouz and Bennouna 2018, Qu and Lemhadri 2021], and finance [Kirchner 2017, Simon 2016]. To merge the advantages of both the Agent-Based Model and Temporal model, this paper will introduce a new simulation approach that implements both intensity- and cluster-based approaches from Hawkes Point Process, incorporating the concept of user networks from Agent Based model where each user will be considered as an agent with proper user behaviors and interactions with other users within the network.

3.3 Preliminaries

The simulation process follows the basic idea about the cluster- and intensity-based simulation methods, but is adapted to the information dissemination process on Twitter: different tweet types and the content will affect the information dissemination rate to varying degrees. Moreover, because the simulation process will focus on fake news, the user stances and networks will influence which users receive the information regarding their opinions towards the fake news. More information about the process

will be introduced in the following subsections.

3.3.1 Twitter

Twitter as a popular online social network, provides a platform for people to share and spread information, which includes regular information, misinformation, and disinformation. Different users may hold different attitudes toward fake news stories, this is what we called '*user stances*'. For instance, users may believe or not believe what fake news says. In addition, the Twitter social network allows the information to be spread among users such that a user's stance may hold influence on other users' opinions towards the fake news, and *user networks* determine the information a user may receive from Twitter such that a user will see the tweets of the users he/she follows, as well as the replies of those tweets. Twitter has multiple *tweet types* to carry the information: original tweets, retweets, quotes, and replies. Particularly, original tweets are the initial tweets that are generated by the user (may come from other platforms or sources), and quotes are a special type of retweets with additional comments from users.

3.3.2 Assumptions

The following are key assumptions and restrictions the simulation process follows.

Assumption 1. The tweets of the original sources of the information with no additional comments should hold a supporting stance as the sources.

These tweets are 'original tweets', and correspond to 'immigrants' in the Hawkes Point Process terminology. Since no additional information is added by the users,

these tweets should hold the same stance as the information sources regardless of whether it is fake news or fact-check. Original tweets *with comments* from users will reveal users' opinions towards the information sources (fake news articles or fact-check articles) such that may hold different stances against the sources.

Assumption 2. The stance of a retweet should be the same as the stance of the event that triggers it.

Retweets are the forwarding of a tweet, unaltered and with no additional information, to the user's network. Because no additional information is supplied in a retweet, the stance and text of a retweet should be identical to the stance and text of the event that triggers it.

Assumption 3. A user who retweeted, quoted, or posted an original tweet of a fake news story in a specific stance will not retweet or quote the same piece of information in the same stance.

This assumes that a user who is interested in the fake news and disseminated the information through an original tweet, retweet, or quote in a specific stance will no longer be interested in generating a similar tweet in the same stance. This assumption restricts the possibility of generating numerous retweets and quotes in the same stance in the simulation process. This assumption does not restrict the users' behaviors of:

- Generating replies in the same stance: Twitter users may create a conversation and discuss the information with others.
- Generating retweets/quotes in other stances: during the dissemination process of fake news stories, fake news will be debunked after a period of time through

fact-check websites. Thus, users may change their minds after reading the fact-check articles, and retweet/quote the information with a different stance.

Assumption 4. Only the users who follow the user that starts the conversation can see the replies under the discussion.

This assumption is determined by the dissemination pattern of Twitter. Consider the user relationships presented in Fig. 3.1A where user A follows user B, and users B and D follow user C. When user B tweets a general tweet such as an original tweet (shows in Fig. 3.1B), retweet, or a quote, all the users who follow user B (e.g., user A) will receive this information and have a chance to be influenced by it. However, if user C tweets a general tweet that starts a conversation between user C and user B with their replies, then only the users who follow user C will observe the replies. This indicates that even though user A follows user B and can see user B's general tweets, user A cannot see user B's replies to other users' tweets (such as user C's tweets) unless user A follows these users as well.

The user network engaged in the process determines the path of information dissemination, which distinguishes the dissemination process on Twitter from other online platforms.

Assumption 5. The distribution of immigrants follows a truncated exponential distribution.

All the retweets, quotes, and replies are initially triggered and generated from the original tweets (immigrants). The original tweets initiate the dissemination process and take a dominant position in the beginning, but are no longer generated after a period of dissemination which gradually leads to the ending of the process. We used a truncated exponential distribution (TruncExpo) to generate the immigrants

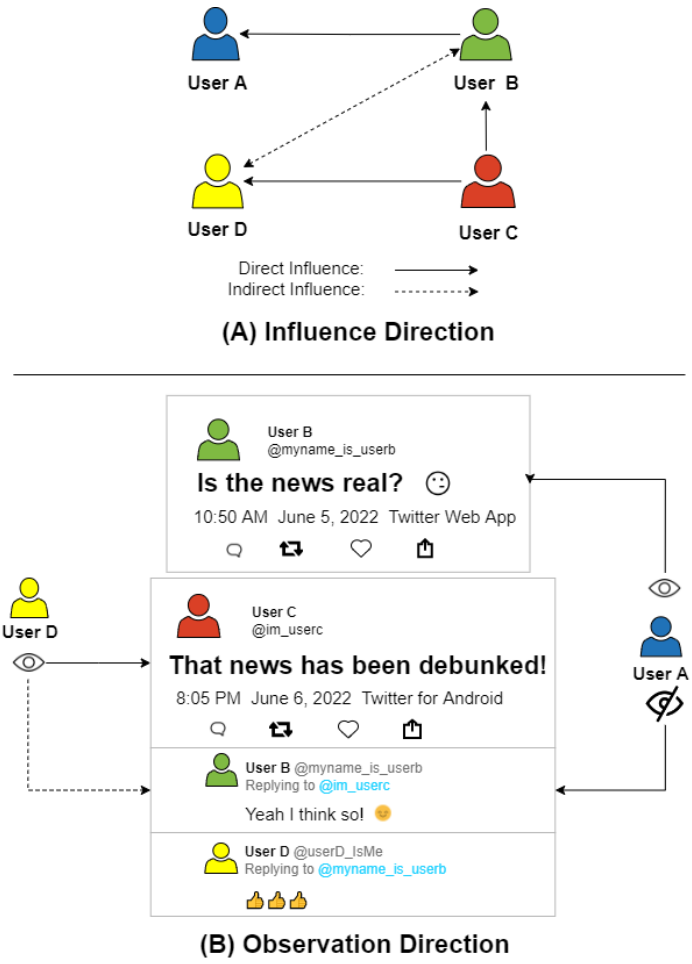

Figure 3.1: Example of A Twitter User Network

which produces a right-skewed pattern that was observed in the real dataset. More information on the real dataset collected will be found in the simulation example section.

3.4 Simulation Description

The simulation combines the cluster- and intensity-based approaches. A cluster-based approach is first used to generate the immigrants and then an intensity-based

approach is used within an agent-based model to generate subsequent events.

3.4.1 Basic Concepts

1. Users: Each user, i or $u_j, (i,\ u_j = 1, 2, \ldots, U)$, can generate a tweet, retweet, quote, or reply given the influence received from the tweets of his/her friends, which are the users that he/she follows. However, there exists the possibility that a user will be influenced by the tweet of a user whom he/she did not follow, such as replies you may see under your friend's tweet but the replies come from someone who follows your friend but did follow by you. In order to capture the information dissemination process thoroughly, such user relationships within the user network should be captured as well.

2. User Networks: $G = (V, E)$ is a directed graph that contains the relationship among Twitter users. A directed edge between two users indicates that one follows the other. A follower receives all the tweets from their connections. Fig. 3.2 shows a simple example with four users $V = \{1, 2, 3, 4\}$. The directed edges, $E = \{(2, 1), (4, 1), (3, 2), (4, 2)\}$ indicate that user 1 follows users 2 and 4, user 2 follows users 3 and 4, but users 3 and 4 do not follow anyone. A user can be directly influenced by the users they follow. A user can also be indirectly influenced by users they don't directly follow. Fig. 3.1A shows a simplified network where user A follows B, users B and D follow C. While users B and D do not directly follow each other, they can still be exposed to some of their tweets because they both follow user C. Fig. 3.1B shows a small tweet cascade where user C makes a tweet that is replied by user B. User D is able to see this reply and further replies to B's message. We capture these indirect connections in our simulation.

3. Tweets: the jth event Z_j is a tweet of type r_j from user u_j with stance k_j.

   - Tweet type r: There are four types of tweets on Twitter which can be summarized as:
     $r \in R = \{ori,\ ret,\ quo,\ rply\}$ representing original tweet, retweet, quote, and reply.
   - Tweet stance k: the stance of the current tweet towards a fake news story, which is not associated with the user but the tweet since users may change their stances over time. Moreover, we will also use k_j to represent the stance of event j. There are two tweet stances that can be summarized as:
     $k \in K = \{s, d\}$ representing supporting and denying stance.
   - Conversation C_j: a conversation that starts from a tweet or quote.
     For example, $C_1 = \{1, 2, 3, 5\}$ refers to the case that conversation C_1 starts from a tweet with tweet id = 1 and it contains tweets with tweet id = 2, 3 and 5.
   - Event set X: a set of events (tweets).
     Particularly, the set of immigrants will be denoted by X_{imm}, and the set of descendants will be denoted by X_{des}.

3.4.2 Model for event intensity

The overall intensity of the fake news dissemination process on Twitter can be expressed as follows:

$$\lambda(t;\theta) = \sum_{k \in K} \left(\mu_k(t) + \sum_{r \in R} \sum_{i}^{U} \sum_{j:t_j<t} \lambda_{kri}(t - t_j; \theta, Z_j) \right) \tag{3.1}$$

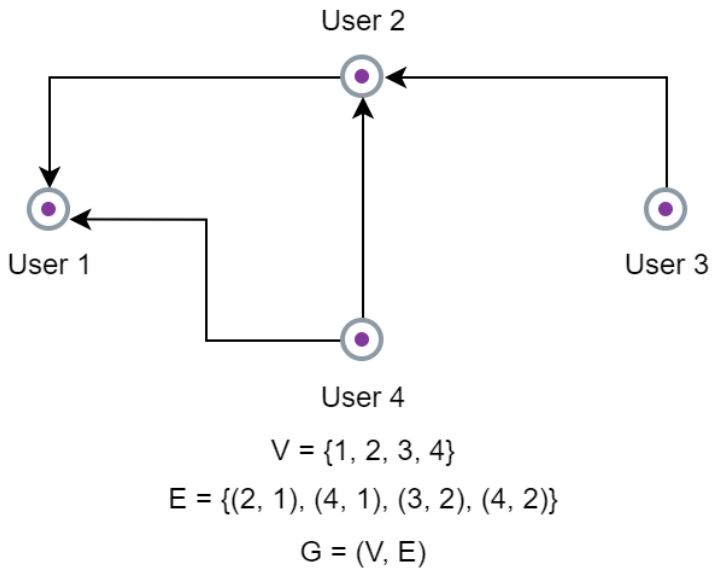

Figure 3.2: Example of An User Network

where t is time and θ are all the model parameters. The term μ_k represents the immigrant rate of original tweets with stance k. The $\lambda_{kri}(t - t_j; \theta, Z_j)$ denotes the event intensity derived for user i generating a tweet type r and stance k at time t given a triggering event $Z_j = \{t_j, k_j, r_j, u_j\}$. The triggering event information is time t_j, stance k_j, type r_j, and user u_j (more details are provided below). The triggering function is parameterized as:

$$\lambda_{kri}(t - t_j; \theta, Z_j) = \delta_{r_j} \beta_{i,u_j} \gamma_{k_j,k} g_k(t - t_j) p_{ki}(r) \tag{3.2}$$

where the parameters are described in the following section.

3.4.3 Model Parameters

1. Immigrant rate $\mu_k(t)$

The arrival rate of immigrants (i.e., original tweets about the fake news event) with stance k. Since there are two stances, the arrival rate of events in each stance will be assigned with its own values: $\mu_s(t)$ and $\mu_d(t)$, referring to two different arrival rates of immigrants. We model the immigrant arrival rate with a truncated exponential density (TruncExpo)

$$\mu_k(t) = \mu_k \cdot \frac{xe^{-xt}}{1 - e^{-xT}} \tag{3.3}$$

where μ_k is the total expected number of immigrants with stance k, x is the scale parameter controlling the skewness of the event distribution, and T is the upper bound on when immigrants can arrive.

2. Influence factor of different tweet types δ_r

   The influence rate of different tweet types that control the intensity that triggers the new events (tweets) to be generated.

3. User relationship factor β_{i,u_j}

   The relationship parameter between user i and the user of event j denoting by u_j. As introduced in the previous section, there are two types of user relationships: direct relationship β_d and indirect relationship β_i. Since users are influenced by the other users whom they follow, a higher value will be set to the relationship factor $\beta_{i,u_j} = \beta_d$ with a direct relationship, and a relatively lower value for an indirect relationship $\beta_{i,u_j} = \beta_i$. The sum of these two values should equal 1, representing the probabilities of being affected by direct and indirect relationship sum to 1.

4. Influence factor between stances $\gamma_{k_j,k}$

The influence rate between two stances towards fake news, measuring the influence that a tweet with stance k_j has on generating a tweet of stance k. There are four parameters to capture the possible interactions of two types of stances.

5. Kernel function $g_k(t - t_j)$

   The kernel function in the terminology of Hawkes Point Processes controls the amount of time a previous tweet can influence users. We used an exponential kernel which dictates that a tweet's influence will diminish exponentially over time. Specifically we use $g_k(t - t_j) = \omega_k e^{-\omega_k(t-t_j)}$ where ω_k is the decay parameter for stance k.

6. Tweet type $p_{ki}(r)$

   The parameters $p_{ki}(r) = p(r|i, k)$ are the probability that user i, holding stance k, will make a tweet of type r. Generally, there are only three types of tweets that will be triggered (retweets, quotes, and replies), therefore:

$$\sum_{r \in R} p(r|i, k) = p(ret|i, k) + p(quo|i, k) + p(rply|i, k) = 1 \tag{3.4}$$

   In addition, $p(ret|i, k)$ and $p(quo|i, k)$ will be changed according to **Assumption 3** during the simulation process since they are dependent on the events that the user has generated prior to the current time t.

Overall, the process is initiated by original tweets (immigrants) but continues as users are influenced to make additional tweets. User activity is highly associated with user networks in terms of the number of other users the current user follows, tweet information, tweet type, and tweet stance of all prior tweets the current user may observe. Such influence will be reflected in the tweet the user will generate

which will influence the next user based on his/her user networks iteratively, creating a self-exciting process within a specific user network. After a specific time point T when no additional immigrants can be generated, the intensity will start to diminish and no more tweets will be generated eventually.

3.4.4 Simulation Process

This simulation process is developed based on both the intensity- and cluster-based simulation approach for Hawkes Process modeling which engages the pre-defined parameters to adapt the dissemination process of fake news on Twitter. One can recognize and differentiate original tweets (immigrants) from all other tweets (descendants), and all descendants are triggered by immigrants initially, such that the process can be described as two parts: the simulation of immigrants using the cluster-based approach, and the simulation of descendants using the intensity-based approach. In order to adapt the information dissemination process on Twitter, a new simulation approach is developed in this paper based on Ogata's thinning method (intensity-based approach) for generating the descendants where a cumulative intensity is considered and calculated on events from all possible users, tweet types, and stances. The overall intensity defined in this paper fits better to the reality, referring to the case that all the users within the Twitter user networks could possibly observe the fake news story propagating on Twitter and make a tweet regarding a possible stance and tweet type with a certain individual intensity value, the higher the intensity the more possible that a tweet with a specific combination of the user, tweet type and tweet stance will be generated.

More details will be discussed as follow.

Simulation of Immigrants

a. Let $t = 0$ be the start of the simulation, $t = T$ to be the end of the simulation, and $t = \frac{T}{x},\ x \in (1, \infty]$ as the point after which the immigrants will no longer be generated.

b. Let μ_s and μ_d be the arrival rates for original events (immigrants) with the stance of supporting and denying.

   - For each stance, generate the number of events from a Poisson distribution (Pois) with mean $= \mu_k T$.

     Generate arrival times for all events of a certain stance following truncated exponential distribution (TruncExpo) where the lower bound, upper bound, and scale are set to be 0 and T, and $\frac{T}{x},\ x \in (1, \infty]$, such that the shape parameter $b = \frac{\text{upper}-\text{lower}}{\text{scale}} = \frac{T-0}{T/x} = x$.

c. Sort X_{imm} by arrival time.

Simulation of Descendants

a. Set the counter of event trails $j = 0,\ j \in [1, N]$ for the event that has been successfully generated. Meanwhile, set another counter index = 0 as the event number that the process will read in the current iteration. Let $t = 0$ and $t = T$ as the start and the end of the simulation.

b. While $t \leq T$:

   i. Let $t = t_{\text{index}}$

ii. Calculate intensities at time t, for all users, all tweet types and stances on the new event. Specifically, it can be decomposed into several steps sequentially:

- Users

  The prior influential events will be aggregated for each user respectively based on their user networks. Thus, we will know what event(s) could affect the user to generate a new event. **Assumption 4** should be considered in here accordingly: any user within the user networks will be influenced by the tweets generated or propagated by the users he/she follows, plus the replies under those tweets.

- Tweet types and stances

  After confirming the events that influence the users, all possible tweet types along with the tweet stances that a user would generate should be considered. As aforementioned, possible types of new tweets include retweets, quotes, and replies with determined probability distribution $p_{ki} = p(r|i,k)$ to generate them. Stances will be considered as well, determined by the parameter $\gamma_{k_j,k}$ based on the stance of the prior tweets k_j. **Assumption 3** will be followed, which means a user will not generate a retweet/quote with a certain stance if he/she has generated one with the stance before. Moreover, **Assumption 2** will be followed, indicating that a possible retweet should hold the same stance as its original tweet.

- Kernel function

  After confirming all possible tweet types and tweet stances that each user will generate for the new tweet, kernel functions for each possible

tweet should be calculated based on all prior tweets that each user will be influenced by, referring to the decaying influence that each user will receive from prior events in his/her user network.

Therefore, the intensities $\lambda_{kri}(t - t_j; \theta, Z_j)$ of user i generating a tweet in type r with stance k will be calculated at this stage for all users, tweet types and stances through equation (3.1).

iii. Calculate the overall intensity λ^* at current time t by summing up the intensities for all users with all tweet types and tweet stances by: $\lambda^* = \lambda(t; \theta)$ through equation (3.1).

iv. Simulate a possible inter-arrival time τ for the new event by:

- Draw $u \sim \text{Unif}(0, 1)$ (uniform distribution between 0 and 1)
- Let $\tau = -\frac{ln(u)}{\lambda^*}$ such that τ follows exponential distribution

v. Update the current time $t = t + \tau$

vi. Calculate the overall intensity $\lambda(t; \theta)$ at current time t as λ by repeating step i. and ii.

vii. Make a decision on accepting or denying the new event by:

- Draw $p \sim \text{Unif}(0, 1)$
- Calculate $\frac{\lambda}{\lambda^*}$ and compare it with p:
  - Accept the new event if $p \leq \frac{\lambda}{\lambda^*}$

    Calculate the probability of generating a candidate event by user i in tweet type r with tweet stance k for all possible combinations of users, tweet types and tweet stances through $p(i, k, r) = \frac{\sum_{t_j < t} \lambda_{kri}(t - t_j; \theta, Z_j)}{\lambda}$

Random sampling through all possible $p(i, k, r)$ and save the selected event into the process as a successful event. Let $j = j + 1$

– Otherwise, reject the sample event and save the event as a rejected event in the process.

viii. Sort X by arrival times through ascending order.

ix. Let index = index + 1 and return to the step i.

3.4.5 Pseudo Code

This section introduces the pseudo code for the simulation algorithm, and the pseudo code for immigrants and descendants will be introduced separately.

Algorithm 1 Simulation of Immigrants

1: Set μ_k for $k \in K$. Set $i \in [1, U]$. Set lower bound = 0, upper bound = T and scale = $\frac{T}{x}$, $x \in (1, \infty]$ such that shape parameter $b = \frac{\text{upper}-\text{lower}}{\text{scale}} = \frac{T-0}{T/x} = x$
2: **for** each stance $k \in K$ **do**
3: Generate $N_k \sim \text{Pois}(\mu_k T)$
4: **for** each event j, $j \in [1, N_k]$ **do**
5: Generate $t_j \sim \text{TruncExpo}(b = x)$
6: Generate u_j for each event j where $u_j \in [1, U]$
7: **end for**
8: **end for**
9: Sort X_{imm} by time in ascending order

3.5 Simulation Example

In this section, a simulation example will be introduced based on a given test user network. This simulation approach can be performed on different online social net-

Algorithm 2 Simulation of Descendants

1: Set $j = 0$ for descendant counter. Set index $= 0$ as the event counter that counts the simulation iteration, which is also the event number that the process will read in the current iteration. Set $t = 0$ and $t = T$ as the start and end of the simulation for descendants. Set δ_r for $r \in R$. Set $\gamma_{k_j,k}$ and ω_k for $k, k' \in K$. Set β_d, β_i. Set $p(ret|i,k) = p_{ret}$; $p(quo|i,k) = p_{quo}$,; $p(rply|i,k) = p_{rply}$ as the initial conditional probability of generating a tweet in tweet type r given the user i and stance k, and set $p(ret|i,k) = p(quo|i,k) = 0$ for the corresponding user i and stance k according to **Assumption 3**.
2: **while** $0 \leq t \leq T$ **do**
3: $t = t_{\text{index}}$
4: **for** each user i, $i \in [1, U]$ **do**
5: **for** each event j, $t_j < t$ **do**
6: $\delta = \delta_{r_j}$
7: **if** $((r_j \neq rply) \bigcap (u_j \in S_i)) \bigcup ((r_j = rply) \bigcap (j \in C \text{ where } u_C \in S_i))$ is True **then**
8: $\beta_{i,u_l} = \beta_d$
9: **else**
10: $\beta_{i,u_l} = \beta_i$
11: **end if**
12: **for** each stance k for the new event, $k \in \{s, d\}$ **do**
13: $\gamma = \gamma_{k_j,k}$, $\omega = \omega_k$
14: Calculate $\lambda_{kri}(t - t_j; \theta, Z_j)$ through equation (3.2). Particularly, $\lambda_{kri}(t - t_j; \theta, Z_j) = 0$ when $r = ret$ and $k_j \neq k$
15: **end for**
16: **end for**
17: **end for**
18: Calculate the overall intensity through $\lambda^* = \lambda(t; \theta)$ through equation (3.1)
19: Draw $u \sim \text{Unif}(0, 1)$, $\tau = -\frac{ln(u)}{\lambda^*}$, $t = t + \tau$
20: Calculate the overall intensity $\lambda = \lambda(t; \theta)$ again at current time t by repeating the steps from step 3 to 18
21: Draw $p \sim \text{Unif}(0, 1)$
22: **if** $p \leq \frac{\lambda}{\lambda^*}$ **then**
23: Accept the event. Calculate the probability for candidate event of each user i, tweet stance k and tweet type r by $p(i,k,r) = \frac{\sum_{t_j < t} \lambda_{kri}(t - t_j; \theta, Z_j)}{\lambda}$.
24: Draw $p \sim P(i,k,r)$ for randomly sampling from all candidate events with corresponding user i, stance k and tweet type r, and save the selected event as the new event with arrival time t
25: $j = j + 1$

```
        if r ∈ {ret, quo} then
            p(ret|i, k) = p(quo|i, k) = 0 according to Assumption 3
        end if
    else
        Reject the event and save the event as a rejected event with time t
    end if
    Sort event set X by event time
    Let index = index + 1
end while
```

works (including the real user network in the Twitter dataset) but we will leave this for readers to practice since the main focus of this paper is not generating networks, but verifying the event simulation algorithm under different user networks. The code for this simulation approach and generating the user network example can be accessed online [Jiang 2022].

3.5.1 User Network

The user network $G = (V, E)$ used in this paper for testing the simulation approach was generated through *NetworkX* [Hagberg, Swart, and S Chult 2008], a python package for network analysis. This user network includes 8000 nodes imitating 8000 users on Twitter, and each node will be connected to 700 nodes on average which is the average number of followers for a Twitter user [Aslam 2022]. Specifically, Erdős-Rényi [Erdős, Rényi, et al. 1960] model will be applied here to generate a random sample user network. In addition, an extra step will be performed to enforce that each node (user) will have at least one edge (relationship) that connects to the user network. The procedures can be summarized as follow:

1) Set the number of users U and an average number of relationships N_{rel} for each user, where the number of relationships should be smaller than the number of

users. In our case, 8000 and 700 are determined for these two terms respectively.

2) Generate the candidate edges $\{(i, i+1), (i, i+2), ..., (i, U)\}$ for each user i. Randomly select one edge from the candidate edges and add it to E to guarantee the connection between the user and the network.

3) For each user, generate a random number between 0 and 1 that follows exponential distribution (Expo) with scale $scale = \frac{N_{rel}}{U} = \frac{700}{8000}$ as the threshold p^*.

4) For each edge (relationship) of the user i, a random probability p that follows uniform distribution (Unif) will be generated and compared with the threshold to determine whether the edge will be added to the set E.

The algorithm to generate a simple user network is described in algorithm 3.

Algorithm 3 Simulation of a Sample User Network

Set appropriate values for U and N_{rel}. Set $E = \{\}$ for edges set.
for each user i, $i \in [1, U]$ **do**
    Generate candidate edges $\{(i, i+1), (i, i+2), ..., (i, U)\}$
    Randomly sample one edge from candidate edges and add it to E
    Generate the threshold $p^* \sim$ Expo(scale $= \frac{N_{rel}}{U}$)
    for each edge **do**
        Draw $p \sim$ Unif$(0, 1)$
        if $p \leq p^*$ **then**
            Accept the edge and add it to E
        end if
    end for
end for

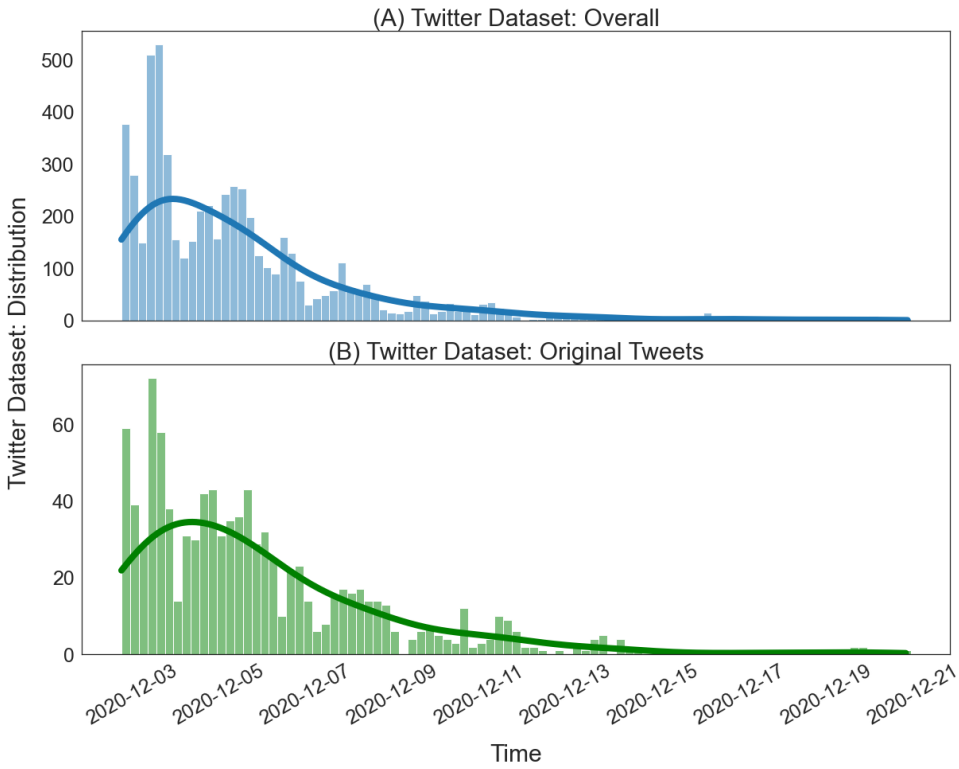

Figure 3.3: Distribution: Twitter Dataset

3.5.2 Parameters

In order to obtain a better combination of values for the simulation parameters, we collected a real Twitter fake news dataset as guidance for the parameter-tuning process of our simulation approach. This dataset was collected based on a piece of fake news that was spread in December 2020 which claimed that the Covid vaccine is female sterilization and later has been debunked by fact-check websites [O'Rourke 2020]. It contains 5909 tweets in total which expand from December 2nd to December 21st, 2020, which shows a decaying trend over time in Fig. 3.3A. The distribution of the original events in Fig. 3.3B suggests a decaying trend as well, such that the truncated exponential distribution (TruncExpo) has been considered for the simulation of immigrants. Based on our observation, we have tested multiple combinations

of parameters and will present the set of values in Table 3.1 which yields the following simulation result. Note that $\gamma_{ss} + \gamma_{sd} = 1$, $\gamma_{ds} + \gamma_{dd} = 1$, $\beta_d + \beta_i = 1$, and $p(ret|i,k) + p(quo|i,k) + p(rply|i,k) = 1$.

Considering the case that the original tweets will trigger all the following events, the value for δ_{ori} was set relatively higher compared to other tweet types; retweets take the majority of the descendants, followed by replies, and quotes are the least, so we set $\delta_{ret} > \delta_{rply} > \delta_{quo}$ and $p(ret|i,k) > p(quo|i,k) = p(rply|i,k)$ for similar reasons.

The values for stance factor γ are $\gamma_{ss} = 0.9$, $\gamma_{sd} = 0.1$, $\gamma_{ds} = 0.5$ and $\gamma_{dd} = 0.5$ which implies that tweets holding a supporting stance will more likely trigger supporting tweets towards the fake news while tweets with denying stance are half as likely to trigger tweets with the same stance.

Larger values for the decay parameter ω for both supporting and denying stances will lead to a fast decay of the influences received from prior tweets, referring to the case that online users will focus more on the recent information instead of tweets posted in the far past. Since we are using a relatively smaller user network with 8000 users who are tightly connected, a fast decaying speed will avoid the case of the supercritical regime [Rizoiu et al. 2017] and the number of events will be bounded.

3.5.3 Simulation Result

Fig. 3.4 shows the simulation result within the small user network with 8000 users in it, where Fig. 3.4A presents the overall distribution, which presents a similar trend compared with the overall trend of the real Twitter dataset in Fig. 3.3A. Fig. 3.4B presents the distribution over tweet types in which *o*, *ret*, *rply* and *quo* in the legend represent the original tweets, retweets, replies, and quotes correspondingly;

Table 3.1: Simulation Parameters

Parameter Names		**Parameter Values**
Simulation Time t	t_0	0
	T	6000
Immigrant Rate μ_k	μ_s	0.15
	μ_d	0.015
TruncExponential Distribution	lower bound	0
	upper bound	6000
	scale x	1000
	shape b	6
Influence Factor of Tweet Type δ_r	δ_{ori}	1.5×10^{-3}
	δ_{ret}	2×10^{-5}
	δ_{quo}	2.5×10^{-6}
	δ_{rply}	5×10^{-6}
Influence Factor between Stances $\gamma_{k',k}$	γ_{ss}	0.9
	γ_{sd}	0.1
	γ_{ds}	0.5
	γ_{dd}	0.5
User Relationship Factor β_{i,u_j}	β_d	0.95
	β_i	0.05
Decay Parameter ω_k	ω_s	3
	ω_d	1.5
Probability of generating tweet in a specific type $p(r\|i,k)$	$p(ret\|i,k)$	0.8
	$p(quo\|i,k)$	0.1
	$p(rply\|i,k)$	0.1

the third plot shows the distribution over tweet stances with supporting and denying attitudes towards fake news. The curves present the smoothed density curve of the corresponding distribution, from which we can find that the retweets count in the second plot increases sharply and goes below the original tweet curve after around $t = 400$, while reply and quote count follow similar distribution but almost overlap with each other since similar parameter values were set for these two tweet types. Fig. 3.4C shows the distribution of supporting and denying tweets over time where the number of supporting tweets is much larger than the number of denying tweets. The overall distribution approximately follows the nature of how tweets are generated on Twitter.

This also can be observed from Table 3.2 showing the total count for each category. To verify the validity of the simulation approach, we can calculate the expected proportion of tweets under different categories and compare them with the results. Recall that we have $\mu_s = 0.15$, $\mu_d = 0.015$ for the immigrant rate of supporting and denying tweets which take about $\frac{\mu_s}{\mu_s+\mu_d} \approx 90.9\%$ and $\frac{\mu_s}{\mu_s+\mu_d} \approx 9.1\%$ respectively, and the count of simulated original tweets (immigrants) in supporting and denying stances are 895 and 116, which take about $\frac{895}{895+116} \approx 88.5\%$ and 11.5% over all simulated original tweets, which are close. The expected proportion of descendants in supporting and denying stances can also be calculated roughly according to the conditional probability: the supporting descendants take about

$$\frac{\mu_s}{\mu_s + \mu_d} \cdot \gamma_{ss} + \frac{\mu_d}{\mu_s + \mu_d} \cdot \gamma_{ds} \approx 86.4\% \tag{3.5}$$

and the proportion of denying descendants is

$$\frac{\mu_s}{\mu_s + \mu_d} \cdot \gamma_{sd} + \frac{\mu_d}{\mu_s + \mu_d} \cdot \gamma_{dd} \approx 13.6\% \tag{3.6}$$

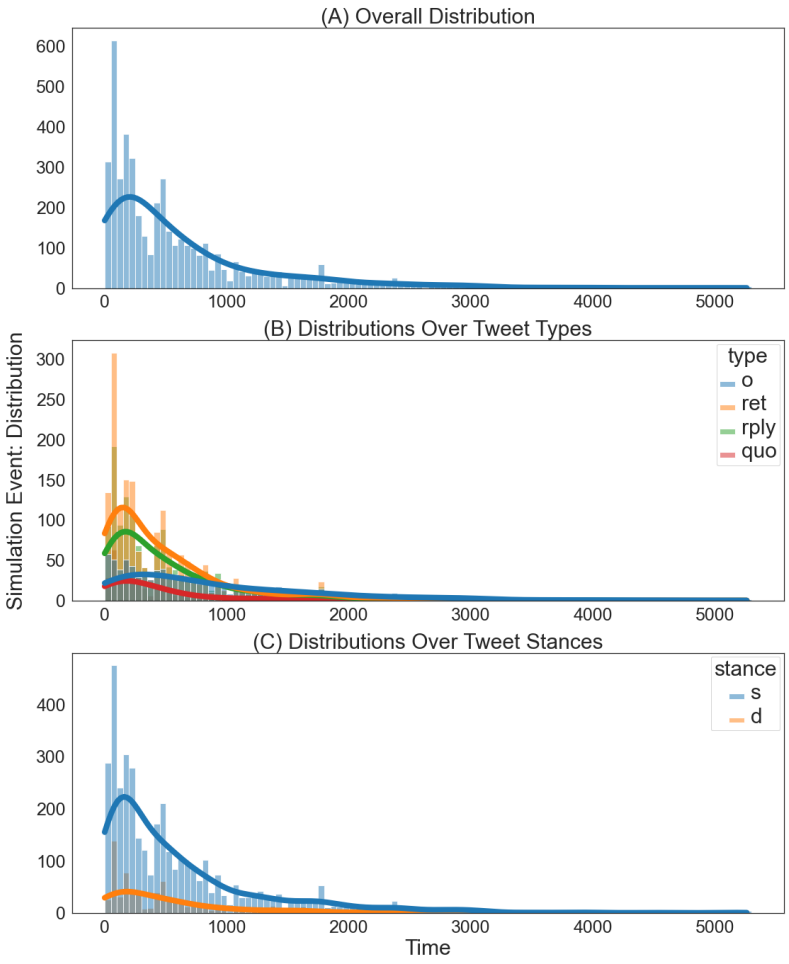

Figure 3.4: Distribution: Simulation

According to the results, the number of supporting descendants (including supporting retweets, quotes, and replies) is $1527 + 231 + 272 = 2030$ which takes about $\frac{2030}{1818+297+344} \approx 82.6\%$, and denying descendants take about 17.4%, which are close to the expected proportions.

Furthermore, the proportion of retweets, quotes, and replies are $\frac{1818}{1818+297+344} \approx 73.9\%$, $\frac{297}{1818+297+344} \approx 12.1\%$, and $\frac{344}{1818+297+344} \approx 13.9\%$ which approximately follows the distribution of the probability of generating retweets (0.8), quotes (0.1), and replies (0.1). Therefore, the overall simulation performance meets our requirements.

Table 3.2: Simulation Event Count

		Tweet Types r				
		Original	**Retweet**	**Quote**	**Reply**	**Total**
Tweet Stance k	**Supporting**	895	1527	231	272	2925
	Denying	116	291	66	72	545
	Total	1011	1818	297	344	3470

3.6 Discussion

This paper proposes a novel simulation method that imitates the dissemination process of fake news on Twitter by modeling through the Multivariate Hawkes Processes model, which incorporates the influence of tweet types, from different user stances toward fake news and the user networks. Furthermore, our simulation approach captures the characteristics of the information dissemination process on Twitter described in **Assumptions 1, 2, and 4**.

There are still remaining limitations of our research work, which can be summarized into the following categories.

3.6.1 User Networks

From the perspective of user networks, there exists the case that user A follows or unfollows user B after user A retweets user B's tweet during the process, which leads to the dynamic of user networks over time and inaccuracy when calculating the influence from the user network. This indicates that a fixed user network from first to last may not capture the whole picture of such an information dissemination process in OSNs, but the real-time user following and follower lists should be considered.

3.6.2 Process Modeling

Another limitation comes from the ability to capture the information dissemination process on Twitter precisely. Twitter and other popular OSNs provide the hashtag function for users to track posts related to a specific topic or keywords such that users are able to access the posts containing the hashtag only. In addition, influence related to review text has been simplified to 1 for all tweet j, which can be expanded to the sentiment score or other measurement that describes the ability to affect tweet behaviors from tweet content. Moreover, supporting and denying stances may not cover all stances since users may become hesitant and interrogative towards the fake news story, or even talk about irrelevant content which does not reflect any specific emotion towards the information.

Therefore, our future work will focus on incorporating the hashtag function in our model, as well as imitating the user behaviors of follow/un-follow fake news disseminators and considering the text factor from tweet content as well as hashtags during the fake news spreading process.

Chapter 4

Modeling and Parameter Estimation of Fake News Dissemination with Multivariate Hawkes Processes

4.1 Introduction

Fake news has never become a novel topic, especially for the online platform in the era of the internet. Fake news, rumors, and information hoaxes are different types of false information, where researchers may provide various definitions for these terms. Although researchers hold different definitions of this term, all of them are in general resemblance with minor differences. A general definition of fake news, according to a recent research [Allcott and Gentzkow 2017], is: "News articles that are intentionally and verifiably false, and could mislead readers".

In the pre-internet era, the consequences of fake news spreading were profound and far-reaching already, and such consequences became even more serious with the development of various means of information dissemination [Burkhardt 2017]. Nowadays, social media and online platforms have become the main source of information dis-

semination [Tandoc Jr, Lim, and Ling 2018]. The varying extent of misinformation will lead to consequences in varying degrees: people who receive inaccurate information will become uncertain about the validity of the knowledge they should be confident with [Rapp and Salovich 2018]. For instance, fake news is always closely related to politics: after the 2016 presidential election, there is still a large portion of people who believe that Clinton's Pizzagate Scandal associated with the child-sex ring was 'probably' or 'definitely' true [Tsfati et al. 2020]; Moreover, fake news has led to the incitement of violence in Nigeria and Nepal [Network 2016]. Prior studies have focused on rumors detection on social media [K. Zhou et al. 2019], fake news detection on Twitter and Facebook using classification methods [Helmstetter and Paulheim 2018, Y. Liu and Y.-F. Wu 2018, Granik and Mesyura 2017], and influence of fake news [Bovet and Makse 2019]. Other relevant studies include fake account detection [Boshmaf et al. 2015] and detection of social bots [Shao, Ciampaglia, et al. 2018]. However, studies have hardly focused on the fake news dissemination itself, which motivates us to explore the possible dissemination patterns of fake news.

User conversations occurring in social media are always accompanied by emerging topics, which produce information cascades where users that are connected tightly through user networks will be affected by the information shared with them and propagate that effect to others. This influence will continuously occur among interrelated users until the influence of the topic slowly decays or a new topic emerges. The huge number of online users and complex user networks make the dissemination process difficult to understand. However, it is necessary to be studied: if the discussion of a topic by online users is defined as an event, the modeling of information cascades will help us understand the trend of event development and can be used

to predict impacts such as event popularity. Studying user feedback and interaction can help us understand user behavioral patterns and influence patterns, which will benefit platform user management and provide users with better services.

The emergence of fake news topics on social media will always initiate heated discussions among users holding different stances, which may lead to social events or even violence, and can have far-reaching impacts on society. Generally, a piece of fake news is generated by fake news websites or blogs and shared to social media platforms such as Twitter and Facebook by website users, and spread by platform users for a certain period. A piece of fake news on micro-blogging websites was spread within hours and led to mass panic and confusion [Islam, Muthiah, and Ramakrishnan 2019]. Based on the fact that opinion leaders and trendsetters exist on social media and take their effects on influencing and leading the public's opinion, it is reasonable to infer that a certain number of online users will be influenced and disseminate such "influence" to the online community. According to the discovered behavioral patterns, there exist user groups that stand against the voices of the mainstream and are not easily affected by others. The introduction of fact-check articles to social media may alter the view of a portion of users and lead to the closure of the discussion; or, it may differ from what the majority believes, and extend the event to a larger discussion. The collision of various stances and views may generate new modes of interaction and influence between online users. Under the context of the dissemination of fake news, the research on the behavioral mode and influence of online users can effectively screen out opinion leaders and trendsetters to restrict and supervise malicious users. At the same time, the platform can target misguided users and aid them, which will contribute to the improvement of social stability. Therefore, it is valuable and nec-

essary to study the information cascade on social media and the interaction between users.

This research focuses on understanding the causal relationship between user stance and responses regarding fake news authenticity and the dissemination process of fake news on social media. Specifically, the following research questions will be addressed regarding fake news dissemination:

1) Does a user's stance toward the veracity of a fake news article affect the dissemination process? If so, how does the stance of a user impact users of different stances? Are there typical interaction patterns between users of different stances?

2) Is fake news dissemination impacted by tweet type? If so, which types of tweets are most influential?

To our best knowledge, this is the first time that user stances and event types have been considered in the Hawkes Process model for analysis of the fake news dissemination problem. A new model is proposed regarding fake news dissemination using the Hawkes Point Process to capture the likelihood of observing upcoming events, which will adapt to the real nature of the fake news dissemination process in online social networks and help answer the research questions. Specifically, parameters associated with user stances and tweet types will be introduced into the model for analyzing their influence on the fake news dissemination process. The paper is organized as follows: prior studies in the related area will be reviewed in section 4.2. Modeling procedures with parameter estimation will be introduced in section 4.3. A real Twitter dataset applied in this paper will be introduced in section 4.4, followed by the

parameter estimation result and findings demonstrated in section 4.5. Limitations and applications will be discussed in section 4.6.

4.2 Literature Review

This section will introduce the prior studies that have been conducted in the area of misinformation/disinformation modeling. Specifically, we will elaborate on this part a general review of fake news dissemination modeling, the information diffusion modeling with user activities using Hawkes Point Processes, and the modeling of fake news dissemination process with Hawkes Point Processes.

4.2.1 Fake News Dissemination Modeling

Fake News Dissemination as a popular research objective, is distinguished from many types of the information diffusion processes. It highlights the importance of user activities and user stances among the social networks, in which the stances the users hold and how users interact with each other will affect users who are involved in the discussion, impact the trend and the consequences of the fake news dissemination. One study has adopted a network simulation model to investigate the possible relationship between echo chamber effects (people prefer to follow like-minded people) and the viral spread of misinformation [Törnberg 2018], which discovered the synergetic effect between opinions and network polarization on the virality of misinformation. Diffusion networks were built with k-core decomposition based on the tweets collected before the 2016 US Presidential Election which found that the core of the network was dominated by social bots while fact-checking almost disappeared

[Shao, Hui, et al. 2018]. A collective influence algorithm in directed networks was developed to uncover how fake news influenced the 2016 US Presidential Election, which found that top influencers spreading traditional center and left-leaning news largely influenced the activities of Clinton supporters while Trump supporters influenced the dynamics of top fake news spreaders [Bovet and Makse 2019]. Fake news modeling can also contribute to the detection of fake news using features of user activities. A fake news propagation model was developed in a related study that divided online users into four types: susceptible, infectious, verified, and recovered which characterized how misinformation disseminated among groups under the influence of different misinformation-refuting measures [Shrivastava et al. 2020]. A prior study that focused on the modeling of fake news spreading on Twitter and Weibo (a Chinese micro-blogging website) found that fake news spreads distinctively from real news events at the early stage, which offered novel features for the early detection of fake news [Z. Zhao et al. 2020].

4.2.2 Hawkes Processes Modeling on Information Dissemination Process with User Activities

Hawkes Point Processes model, as introduced in the previous chapters, has been frequently implemented in the domain of information diffusion. One study developed a Time-Dependent Hawkes process that focused on the temporal patterns of retweet activity of an original tweet, considered the circadian nature of users and the aging information, and performed prediction of the size of the information cascades [Kobayashi and Lambiotte 2016]. A DeepHawkes model was constructed to characterize the information cascades while possessing the predictive power of deep learning to perform prediction on the future popularity of information cascades [Cao

et al. 2017]. A prior study extended the SIR (Susceptible-Infected-Recovered) model with self-exciting processes taken from Hawkes Process and user behaviors to perform prediction on the popularity of information cascades [Kong, Rizoiu, and Xie 2020].

User behavior has hardly occurred individually since people are interacting with each other regarding emerging topics online, and that is how online social networks operate. Based on the discussion in the introduction section, from the perspective of psychology, user stance, reaction, and interaction as cognitive behavior can be explored for understanding the causal relationship between user behaviors or towards the dissemination mechanism and consequences of misinformation. Thus, user behaviors including responses and interactions should be considered in the model to adapt to the real nature of the information dissemination process on online platforms. A Hawkes process model incorporating the user and topic interactions has been created for information dissemination on social networks, in which the model not only derives the influence between users, but also the influence between multiple types of topics [Pinto and Chahed 2015]. This model has been applied in trend detection in social networks [Pinto, Chahed, and Altman 2015], and extended to the circumstance of multiple social networks with user-user, topic-topic, and user-topic interactions [Pinto 2016]. A co-evolutionary latent feature process model that accurately captures the co-evolving nature of users' and items' features was developed for recommendation systems in online service websites, which implements user-item interactions [Yichen Wang et al. 2016]. One prior study built a Fourier-based Multidimensional Hawkes Process to investigate the correlations between online users' activities, which has been evaluated on Github and Metafilter datasets for activity prediction [S. Li et al. 2017].

4.2.3 Hawkes Processes Modeling on Disinformation

In the domain of the dissemination process of disinformation on social media, Hawkes Point Process has been applied frequently which presents its adaptability and practicability in characterizing the dissemination process and predicting the subsequent results. A two-stage Hawkes Process model was built for characterizing the process of fake news dissemination before/after fact-checking occurs [Murayama et al. 2021]. A related study applied the Multivariate Hawkes Process incorporating user networks as a matrix that measures the influence rate between online users to model the process of rumor propagation on Twitter [Nie et al. 2020]. A classification method has been developed for user stances of rumors on Twitter using Hawkes Processes and Maximum Likelihood Estimation [Lukasik et al. 2016]. Similarly, Multivariate Hawkes Process using textual-based base intensity was built for rumor stance classification as well [Tondulkar et al. 2022]. Pathogenic user accounts on social media have been studied with respect to their corresponding user behaviors to analyze the subsequent negative influence using the Hawkes Process model [Alvari and Shakarian 2019]. A hybrid model combining Hawkes Process and Topic Modeling was developed incorporating temporal and textual features for detecting fake retweeters [Dutta et al. 2020]. To combat fake news, the Hawkes Process model was also adapted incorporating reinforcement learning to detect and make interventions in the information diffusion process [Farajtabar et al. 2017, Goindani and Neville 2020].

According to prior studies, user activities and user networks are the key factors that are considered in the modeling of the information dissemination process. Moreover, user stances play an important role in influencing users' opinions during the spreading of fake news. Therefore, a Multivariate Hawkes Point Process is proposed in this study with appropriate parameter adjustment with respect to user stances, user networks,

and different tweet types generated by users on Twitter.

4.3 Methodology

This section will introduce the Multivariate Hawkes Point Processes model that applies to the modeling of fake news dissemination on Twitter in section 4.3.1. The Maximum Likelihood Estimation with the Expectation Maximization algorithm is applied to estimate the model built on the dataset and learn the mechanism of fake news dissemination.

4.3.1 Modeling

This paper implements the model proposed in section 3.4.2 with appropriate simplification and adjustment for the parameter estimation procedures. As aforementioned, the intensity of generating a new tweet can be modeled as a multivariate process. The intensity that a user i generating a tweet with stance k in type r where $r =$ original tweet is:

$$\lambda_{kri}(t - t_j; \theta, Z(t)) = \frac{\mu_k(t)}{U} \tag{4.1}$$

where U is the total number of users in the social networks who engage in the fake news dissemination process, $Z(t)$ is the aggregation of the events (tweets) that generated before current time t, t_j is the arrival time of the tweet j, k is the user stance towards a fake news story which will be illustrated later with details, and $\mu_k(t)$ is the immigrant function that controls the rate of generating immigrants, which are the original tweets

in the Twitter scenario, with the following form:

$$\mu_k(t) = \mu_k \frac{x e^{-xt}}{1 - e^{-xT}} \tag{4.2}$$

It models the process of generating original tweets through a base rate μ_k and a truncated exponential distribution with scale parameter $x,\ 0 < x < T$, and bounded between 0 and total observation time T. The truncated exponential distribution imitates the arrival time of the original tweets with a right-skewed pattern.

The intensity that a user i generating a tweet with stance k in type r where r is retweet, quote, or reply can be expressed:

$$\lambda_{kri}(t - t_j; \theta, Z(t)) = \sum_{j=1}^{N(t)} \delta_{r_j} \beta_{i,u_j} \gamma_{k_j,k} g_k(t - t_j) p_{ki}(r) \tag{4.3}$$

$$= \sum_{j=1}^{N(t)} \delta_{r_j} \beta_{i,u_j} \gamma_{k_j,k} g_k(t - t_j) p_r \tag{4.4}$$

where

1) j denotes the tweet number of a prior tweet.

2) i and u_j $(1 \leq i, u_j \leq U)$ denote the potential user that has been influenced by the prior tweets and the user of event j respectively.

3) $r,\ r \in R$ denotes the tweet types which include original tweets, retweets, quotes, and replies.

4) $k,\ k \in K$ denotes the user stances. This paper considers two possible stances during the dissemination process: (1) supporting stance: the stance of the tweets with obvious support from the users; (2) denying stance: the stance of

the tweets with no obvious support from the users, which may include denying, questioning, or commenting (no stance towards the fake news story).

such that we have the following parameters associated with user relationships, user stances, tweets, and tweet types:

1) δ_{r_j} refers to the influence factor of the tweet type of the prior tweet j

2) β_{i,u_j} refers to the factor of user relationships between user i and user u_j.

3) $\gamma_{k_j,k}$ refers to the influence factor between stance k and stance k_j (stance of event j).

4) $g_k(t-t_j)$ refers to the kernel function that controls the decaying and influencing time of an event.

5) p_r refers to the probability of generating a new tweet in tweet type r

Specifically, we have:

$$\beta_{i,u_j} = \begin{cases} 0.95 & \text{if user i follows the user of event } j \\ 0.05 & \text{if user } i \text{ is able to see the user of replies (event } j\text{) under the tweets} \\ & \text{(including original tweets, quotes, and retweets) from someone user } i \text{ follows} \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

$$p_{ki}(r) = p(r|i,k) = p_r = \begin{cases} p_1 & \text{if the new tweet is a retweet} \\ p_2 & \text{if the new tweet is a quote} \\ 1 - p_1 - p_2 & \text{if the new tweet is a reply} \\ 0 & \text{if the new tweet is an original tweet} \end{cases} \tag{4.6}$$

such that

$$\sum_{r \in R} p_{ki}(r) = \sum_{r \in R} p_r = p_{ret} + p_{quo} + p_{rply} + p_{ori} = 1 \tag{4.7}$$

This indicates that the probability of generating any tweet type follows the distribution in equation 4.6, and the probability of generating an original tweet given prior tweets is 0, which means the emergence of original tweets is not influenced by the self-exciting process.

We further simplify the notation of user relationships as follows:

$$\sum_{i}^{U} \beta_{i,u_j} = n_j \tag{4.8}$$

where n_j is a function that combines the values of user influences between the current user i and each of the users u_j who generated the prior tweet j. Please note that n_j is just a parameter for purpose of notation simplification, and we still need to calculate the intensity of a prior event by multiplying the user relationship value β_{i,u_j} by other parameters according to equation 4.4.

We can aggregate the event intensity to the intensity of a specific stance k and simplify

it through:

$$\lambda_k(t;\theta) = \mu_k(t) + \sum_{r\in R}\sum_{i}^{U}\left(\sum_{j=1}^{N(t)}\lambda_{kri}(t-t_j;\theta,Z(t))\right) \tag{4.9}$$

$$= \mu_k \cdot \frac{xe^{-xt}}{1-e^{-xT}} + \sum_{r\in R}\sum_{i}^{U}\sum_{j=1}^{N(t)}\delta_j\beta_{i,u_j}\gamma_{k_j,k}g_k(t-t_j)p_r \tag{4.10}$$

$$= \mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)}\delta_{r_j}\gamma_{k_j,k}g_k(t-t_j)\sum_{i}^{U}\beta_{i,u_j}\sum_{r\in R}p_r \tag{4.11}$$

$$= \mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)}\delta_{r_j}\gamma_{k_j,k}g_k(t-t_j)\cdot n_j\cdot 1 \tag{4.12}$$

$$= \mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)}\delta_{r_j}\gamma_{k_j,k}n_jg_k(t-t_j) \tag{4.13}$$

$$= \mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)}h_k(t-t_j;\theta) \tag{4.14}$$

such that the corresponding equation to the equation 3.1 can be derived as follow:

$$\lambda(t;\theta) = \sum_{k\in K}\left(\mu_k(t) + \sum_{i}^{U}\sum_{j=1}^{N(t)}\lambda_{kri}(t-t_j;\theta,Z(t))\right) \tag{4.15}$$

$$= \sum_{k\in K}\left(\mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{r\in R}\sum_{i}^{U}\sum_{j=1}^{N(t)}\lambda_{kri}(t-t_j;\theta,Z(t))\right) \tag{4.16}$$

$$= \sum_{k\in K}\left(\mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{r\in R}\sum_{j=1}^{N(t)}\delta_{r_j}\gamma_{k_j,k}n_jg_k(t-t_j)\right) \tag{4.17}$$

$$= \sum_{k\in K}\left(\mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)}\delta_{r_j}\gamma_{k_j,k}n_jg_k(t-t_j)\cdot 1\right) \tag{4.18}$$

$$= \sum_{k\in K}\left(\mu_k\frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)}h_k(t-t_j;\theta)\right) \tag{4.19}$$

where

$$\sum_{j=1}^{N(t)} h_k(t - t_j; \theta) = \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t - t_j) \tag{4.20}$$

refers to the self-exciting function which models the intensity of each of the prior event (tweet) j at time t with respect to the factors (parameters) predefined.

4.3.2 Parameter Estimation

In order to apply the model to a specific dataset to study how the parameters defined in the model impact the dissemination process, parameter estimation procedures should be performed. Maximum Likelihood Estimation is implemented by incorporating the Expectation Maximization approach to recursively optimize the expectation of the log-likelihood function and update parameters. This part will be demonstrated through the following two subsections.

Maximum Likelihood Estimation

Based on the model and the terms defined associated with different factors of fake news dissemination on Twitter, we can calculate the Likelihood Function for the intensity of events in stance k:

$$L_k(t; \theta) = \prod_{j=1}^{N(t)} \lambda_k(t_j; \theta) e^{-\int_0^T \lambda_k(t;\theta) \mathrm{d}t} \tag{4.21}$$

such that the log-likelihood function in stance k can be expressed as follow:

$$\log L_k(t; \theta) = -\int_0^T \lambda_k(t; \theta) \mathrm{d}t + \sum_{j=1}^{N(t)} \log \lambda_k(t; \theta) \tag{4.22}$$

$$= -\int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j; \theta)\right) \mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k(t) + \sum_{l=1}^{N(t_j)} h_k(t_j - t_l; \theta)\right) \tag{4.23}$$

$$= -\int_0^T \left(\mu_k \frac{xe^{-xt}}{1 - e^{-x}} + \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t - t_j)\right) \mathrm{d}t \tag{4.24}$$
$$+ \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{xe^{-xt}}{1 - e^{-x}} + \sum_{l=1}^{N(t_j)} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right)$$

$$= -\mu_k - \sum_{j=1}^{N(t)} \int_{t_j}^{T} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t - t_j) \mathrm{d}t \tag{4.25}$$
$$+ \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{xe^{-xt}}{1 - e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right)$$

Therefore, the log-likelihood function for the overall intensity can be expressed as follow:

$$\log L(t; \theta) = \sum_{k \in K} \log L_k(t; \theta) \tag{4.26}$$

$$= \sum_{k \in K} \left[-\int_0^T \lambda_k(t; \theta) + \sum_{j=1}^{N(t)} \log \lambda_k(t_j; \theta)\right] \tag{4.27}$$

$$= \sum_{k \in K} \Bigg[-\mu_k - \sum_{j=1}^{N(t)} \int_{t_j}^{T} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t - t_j) \mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{xe^{-xt}}{1 - e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right) \Bigg] \tag{4.28}$$

Expectation Maximization

Expectation Maximization algorithm (EM) [Dempster, Laird, and Rubin 1977] is a classical approach for parameter estimation in Hawkes Process models as it finds the

maximum likelihood under the case of unobserved data and latent variables from the dataset. EM algorithm alternates between the expectation step (E-step) and the maximization step (M-step), where the E-step calculates the expectation of the log-likelihood function which is also called the Q function, while the M-step estimates the parameters for the Q function for the next iteration of E-step until the Q function converges. EM algorithm will be illustrated through the following equation derivation on the committed steps, and the complete derivation will be found in Appendix B.

1. Expectation Step

   As aforementioned, E-step calculates the expectation of the log-likelihood function, which is also called the Q function. Based on Jensen's inequality, we performed the following transformation for the log-likelihood function in stance k:

$$\log L_k(t;\theta) = \sum_{j=1}^{N(t)} \log\left(\mu_k(t) + \sum_{l=1}^{N(t_j)} h_k(t_j - t_l;\theta)\right) - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j;\theta)\right) \mathrm{d}t \tag{4.29}$$

$$= \sum_{j=1}^{N(t)} \log\left(p_{jj}^k \cdot \frac{\mu_k(t)}{p_{jj}^k} + \sum_{l=1}^{N(t_j)} p_{jl}^k \cdot \frac{h_k(t_j - t_l;\theta)}{p_{jl}^k}\right) - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j;\theta)\right) \mathrm{d}t \tag{4.30}$$

$$\geq \sum_{j=1}^{N(t)} \left[p_{jj}^k \log\left(\frac{\mu_k(t)}{p_{jj}^k}\right) + \sum_{l=1}^{N(t_j)} p_{jl}^k \log\left(\frac{h_k(t_j - t_l;\theta)}{p_{jl}^k}\right)\right] - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j;\theta)\right) \mathrm{d}t \tag{4.31}$$

$$
\begin{aligned}
&= \sum_{j=1}^{N(t)} p_{jj}^{k} \log(\mu_k \frac{x e^{-xt}}{1-e^{-xT}}) - \mu_k + \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) \\
&\quad - \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t + C
\end{aligned}
\tag{4.32}
$$

where p_{jj} and p_{jl} are the probabilities that event j is an immigrant (original tweet), or it is a descendant (retweets, quotes, or replies) caused by any prior influential tweets respectively.

Thus, Q function in stance k takes the form of equation 4.32:

$$
\begin{aligned}
Q_k(T;\theta) = &\sum_{j=1}^{N(t)} p_{jj}^{k} \log(\mu_k \frac{x e^{-xt}}{1-e^{-xT}}) - \mu_k + \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) \\
&- \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t + C
\end{aligned}
\tag{4.33}
$$

and the overall Q function is expressed as:

$$
\begin{aligned}
Q(T;\theta) = &\sum_{k \in K} \sum_{j=1}^{N(t)} p_{jj}^{k} \log(\mu_k \frac{x e^{-xt}}{1-e^{-xT}}) - \sum_{k \in K} \mu_k + \sum_{k \in K} \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) \\
&- \sum_{k \in K} \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t + C
\end{aligned}
\tag{4.34}
$$

Therefore, the following calculations should be performed in E-step to calculate the p_{jj} and p_{jl} when applying exponential distribution as the kernel function:

$$
p_{jj}^{k\ (s+1)} = \frac{\mu_k^{(s)} \frac{x e^{-x t_j}}{1-e^{-xT}}}{\mu_k^{(s)} \frac{x e^{-x t_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l, k_j}^{(s)} n_l g_k(t_j - t_l)}
\tag{4.35}
$$

$$= \frac{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}}}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j - t_l)}} \tag{4.36}$$

$$p_{jl}^{k\,(s+1)} = \frac{\delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)} \tag{4.37}$$

$$= \frac{\delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k e^{-\omega_k(t_j - t_l)}}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j - t_l)}} \tag{4.38}$$

2. Maximization Step

   Q function represents the lower bound of the log-likelihood function, and the M-step is to maximize the lower bound by maximizing the values of the parameters such that the algorithm iterates over the E-step and M-step and keeps searching for the optimal solutions until the Q function converges.

   The maximum value of parameter θ occurs when $\frac{\partial Q(T;\theta)}{\partial \theta} = 0$ such that for μ_k in $Q_k(T;\theta)$ we have:

$$\frac{\partial Q_k(T;\theta)}{\partial \mu_k} = \sum_{j=1}^{N(t)} p_{jj}^k \frac{1}{\mu_k \cdot \frac{xe^{-xt}}{1-e^{-xT}}} \cdot \frac{xe^{-xt}}{1-e^{-xT}} - 1 = 0 \tag{4.39}$$

$$\frac{1}{\mu_k} \sum_{j=1}^{N(T)} p_{jj}^k = 1 \tag{4.40}$$

$$\mu_k = \sum_{j=1}^{N(t)} p_{jj}^k \tag{4.41}$$

   Particularly, as the simulation introduced in the section 3.4.4, the number of immigrants is modeled and simulated based on the Poisson distribution Pois with a mean of overall immigrant rate which equals to $\mu_k T$, such that the equation of updating μ_k at each iteration $s+1$ of calculation is equivalent to :

$$\mu_k^{(s+1)} = \frac{\sum_{j=1}^{N(t)} p_{jj}^{k\,(s)}}{T} \tag{4.42}$$

For the other parameters associated with the self-exciting process when applying the exponential kernel function, we have:

$$\frac{\partial\left(\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^{k}\log h_k(t_j-t_l;\theta)\right)}{\partial\theta}-\frac{\partial\left(\sum_{j=1}^{N(t)}\int_{t_j}^{T} h_k(t-t_j;\theta)\mathrm{d}t\right)}{\partial\theta}=0 \tag{4.43}$$

$$\frac{\partial\left(\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^{k}\log h_k(t_j-t_l;\theta)\right)}{\partial\theta}=\frac{\partial\left(\sum_{j=1}^{N(t)}\int_{t_j}^{T} h_k(t-t_j;\theta)\mathrm{d}t\right)}{\partial\theta} \tag{4.44}$$

$$\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^{k}\frac{\partial_\theta\left(\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}\right)}{\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}}=\sum_{j=1}^{N(t)}\partial_\theta\left(\delta_{r_j}\gamma_{k_j,k}n_j\cdot\left(1-e^{-\omega_k(T-t_j)}\right)\right) \tag{4.45}$$

For $\gamma_{k',k}$, we have:

$$\sum_{j=1}^{N(t)}\sum_{l:k_l=k'}^{N(t_j)} p_{jl}^{k}\frac{1}{\gamma_{k',k}}=\sum_{j=1}^{N(t)}\delta_{r_j}n_j\left(1-e^{-\omega_k(T-t_j)}\right) \tag{4.46}$$

$$\gamma_{k',k}=\frac{\sum_{j=1}^{N(t)}\sum_{l:k_l=k'}^{N(t_j)} p_{jl}^{k}}{\sum_{j=1}^{N(t)}\delta_{r_j}n_j\left(1-e^{-\omega_k(T-t_j)}\right)} \tag{4.47}$$

Consider the case that a fake news story outbreaks on Twitter initially, diminishes over time, and vanishes eventually, the occurring time of each event t_j should deviate from the total time T of the observation as the intensity decays over time, such that $e^{-\omega_k(T-t_j)}\approx 0$. Thus, the above expression can be simplified to its closed-form expression:

$$\gamma_{k',k}\approx\frac{\sum_{j=1}^{N(t)}\sum_{l:k_l=k'}^{N(t_j)} p_{jl}^{k}}{\sum_{j=1}^{N(t)}\delta_{r_j}n_j} \tag{4.48}$$

Particularly, as the **Assumption 2** illustrated in section 3.3.2, all the retweets should hold the same stance as the tweet triggers it, such that the between-

stance factor $\gamma_{k',k}$ should take effect on generating quotes and replies with the following adjustment on the expression at each iteration $s+1$:

$$\gamma_{k',k}{}^{(s+1)} = \frac{\sum_{j:r_j\in\{quo,rply\}}^{N(t)} \sum_{l:k_l=k'}^{j-1} p_{jl}^{k\,(s)}}{\sum_{k\in K} \sum_{j:r_j\in\{quo,rply\}}^{N(t)} \sum_{l:k_l=k'}^{j-1} p_{jl}^{k\,(s)}} \tag{4.49}$$

$$\tag{4.50}$$

Similarly, for ω_k, we have:

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \frac{\cdot(1-\omega_k(t_j-t_l))}{\omega_k} = \sum_{j=1}^{N(t)} \delta_{r_j}\gamma_{k_j,k} n_j(T-t_j)e^{-\omega_k(T-t_j)} \tag{4.51}$$

$$\omega_k \left[\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k}(t_j-t_l) + \sum_{j=1}^{N(t)} \delta_{r_j}\gamma_{k_j,k} n_j(T-t_j)e^{-\omega_k(T-t_j)}\right] = \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \tag{4.52}$$

$$\omega_k = \frac{\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k}}{\sum_{j=1}^{N(t)} \left[\sum_{l=1}^{N(t_j)} p_{jl}^{k}(t_j-t_l) + \delta_{r_j}\gamma_{k_j,k} n_j(T-t_j)e^{-\omega_k(T-t_j)}\right]} \tag{4.53}$$

such that we have the following simplified expression for ω_k to be updated at each iteration $s+1$:

$$\omega_k^{(s+1)} = \frac{\sum_{j=1}^{N(t)} \sum_{l=1}^{j-1} p_{jl}^{k\,(s)}}{\sum_{j=1}^{N(t)} \sum_{l=1}^{j-1} p_{jl}^{k\,(s)}(t_j-t_l)} \tag{4.54}$$

Since δ_r is associated with all user stance k but specific to tweet type r such

that we need to derive its expression over the overall $Q(T;\theta)$ function:

$$\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^{k}\frac{\partial_{\delta_r}\left(\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}\right)}{\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{k\in K}\sum_{j:r_j=r}^{N(t)}\partial_{\delta_r}\left(\delta_{r_j}\gamma_{k_j,k}n_j\cdot\left(1-e^{-\omega_k(T-t_j)}\right)\right) \tag{4.55}$$

such that we have:

$$\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^{k}\frac{1}{\delta_r} = \sum_{k\in K}\sum_{j:r_j=r}^{N(t)}\gamma_{k_j,k}n_j\left(1-e^{-\omega_k(T-t_j)}\right)\Big) \tag{4.56}$$

$$\delta_r = \frac{\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^{k}}{\sum_{k\in K}\sum_{j:r_j=r}^{N(t)}\gamma_{k_j,k}n_j\left(1-e^{-\omega_k(T-t_j)}\right)\Big)} \tag{4.57}$$

Hence, the closed-form solution for δ_r at each iteration $s+1$ is:

$$\delta_r^{(s+1)} = \frac{\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{j-1} p_{jl}^{k\,(s)}}{\sum_{k\in K}\sum_{j:r_j=r}^{N(t)}\gamma_{k_j,k}^{(s)}n_j} \tag{4.58}$$

3. Q function

After E-step and M-step at each iteration, the $Q(T;\theta)$ should be updated through the following computation to determine whether it converges:

$$\begin{aligned} Q^{(s+1)}(T;\theta) = &\sum_{k\in K}\sum_{j=1}^{N(t)} p_{jj}^{k\,(s)}\log(\mu_k^{(s)}\frac{xe^{-xt}}{1-e^{-xT}}) - \sum_{k\in K}\mu_k^{(s)} \\ &+ \sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l=1}^{j-1} p_{jl}^{k\,(s)}\log h_k^{(s)}(t_j-t_l;\theta) \\ &- \sum_{k\in K}\sum_{j=1}^{N(t)}\int_{t_j}^{T} h_k^{(s)}(t-t_j;\theta)\mathrm{d}t + C \end{aligned} \tag{4.59}$$

$$
\begin{aligned}
&\approx \sum_{k\in K}\sum_{j=1}^{N(t)} {p_{jj}^{k}}^{(s)} \log(\mu_k^{(s)} \frac{xe^{-xt}}{1-e^{-xT}}) - \sum_{k\in K} \mu_k^{(s)} \\
&+ \sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l=1}^{j-1} {p_{jl}^{k}}^{(s)} \log\left(\delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j-t_l)}\right) \qquad (4.60) \\
&- \sum_{k\in K}\sum_{j=1}^{N(t)} \delta_{r_j}^{(s)} \gamma_{k_j,k}^{(s)} n_j + C
\end{aligned}
$$

We will compare $Q^{(s+1)}$ with $Q^{(s)}$ until it converges:

$$
|Q^{(s+1)}(T;\theta) - Q^{(s)}(T;\theta)| \leq \epsilon \qquad (4.61)
$$

which indicates that the optimal solutions for the parameters have been reached.

4.4 Dataset

This section introduces the dataset we applied to verify the parameter estimation procedures we developed as well as to study the interaction pattern and mechanism of the fake news dissemination process on Twitter.

4.4.1 Simulation Dataset

The simulation dataset applied in this paper is the dataset generated through the simulation approach introduced in section 3.4 in the previous chapter where the dataset is introduced in section 3.5.3 with the parameter values defined in table 3.1. The procedures of implementing a simulated dataset with pre-defined parameter values for the simulation show the necessity of verifying the accuracy of the parameter es-

timation equations derived in the previous section because we will further apply the parameter estimation equations to investigate the real fake news dissemination process on Twitter concerning the influence of tweet types and the influence between users' stances towards the fake news story.

As aforementioned in table 3.2, this simulated dataset contains 3470 simulated events with 2925 supporting tweets, 545 denying tweets, 1011 original tweets, 1527 retweets, 297 quotes, and 344 replies.

4.4.2 Real Twitter Dataset

A real Twitter dataset collected through Twitter API v1 and v2 is applied in this paper to show an example of how fake news disseminates over time on Twitter. This dataset has also been introduced in the previous chapter in section 3.5.3: it contains 5909 tweets that relate to one of the fake news stories that occurred during December 2020 which claimed that Covid vaccine is female sterilization and has been debunked later [O'Rourke 2020], with 4725 supporting tweets, 1184 denying tweets, 993 original tweets, 3895 retweets, 948 replies, and 73 quotes, which is summarized in Table 4.1. All the labels of tweet stances are assigned manually. Note that the denying stance defined in this paper does not represent an obvious denying attitude from the user, but not showing an obvious supporting stance from the tweet content, which includes the case of showing denying stance, questioning the fake news story, and commenting on the fake news without any stance.

In addition, the user relationships were collected for each user that engaged in the propagation of the above fake news story to form the user network and study the information cascades. Based on the dataset, 4577 users were found that generated the tweets associated with the fake news story, however, only 2963 users' relationships

were found and collected through the data collection due to the account suspension and user privacy settings of being able to be accessed through Twitter API. For those tweets in which the user account information is missing, there is no obvious influence direction we could obtain from the user relationships. In such a case, we will rely on the information attached to the tweet of whether it is a retweet/quote/reply of a prior tweet in our dataset, and consider this prior tweet as the influential prior tweet of the current one. If the current tweet is a retweet/quote/reply of a prior tweet that does not belong to our dataset, in other words, an irrelevant tweet towards the fake news story, then we will assume that the user received the information about the fake news story from other sources such as hashtags or keywords, and we will consider all tweets that posted prior than the current tweet as the prior influential tweets.

This dataset has been used and considered as an indicator for the values set for parameters in the simulation process, and the usage of this real Twitter fake news dataset in the current chapter will reveal the answers to the research questions by modeling the fake news dissemination process with the Multivariate Hawkes Processes.

Table 4.1: Real Twitter Dataset: Event Count

		Tweet Types r				
		Original	**Retweet**	**Quote**	**Reply**	**Total**
Tweet	**Supporting**	396	3645	52	583	4725
Stance	**Denying**	597	250	21	365	1184
k	**Total**	993	3895	73	948	5909

4.5 Results

This section will present the results of parameter estimation by applying the EM algorithm on both the simulated dataset and the real Twitter dataset.

4.5.1 Parameter Estimation on Simulated Dataset

Table 4.2 compares the true values and the estimation result for each parameter in the model where the parameters with pre-defined values are not included in the table. As the table shows, the EM algorithm estimates most of the parameters accurately such as the base immigrant rates for both supporting μ_s and denying (not supporting) stances μ_d, the scale parameter x of the truncated exponential distribution, and the influence factor between stances where $\gamma_{ds} = 0.3473$ and $\gamma_{dd} = 0.6527$ slightly deviate from the true values (0.5 and 0.5) due to the very limited number of denying tweets generated in the simulation. The sum of γ_{ds} and γ_{dd}, as well as γ_{ss} and γ_{sd} equal to 1 which matches the meaning we defined for γ as it represents the conditional probabilities of generating supporting and denying tweets given any specific stance should sum up to 1.

The estimation of influence factors of tweet types does not match their true values closely due to the edge effect of setting relatively smaller values for them, and the estimations of decay parameters deviating from the true values may be due to the inaccurate estimations of the influence factors of tweet types. However, the relative relationships between the same parameter type are almost correct which reveals the fact that original tweets take the dominant place in influencing new tweets, followed by retweets and the other two tweet types, and the supporting tweets will hold a longer influence than the denying tweets.

Table 4.2: Parameter Estimation: Simulated Dataset

Parameter Names		**True Values**	**Estimation**
Immigrant Rate μ_k	μ_s	0.15	0.1492
	μ_d	0.015	0.0193
TruncExponential Distribution	lower bound	0	/
	upper bound	6000	/
	scale x	1000	969.9030
Influence Factor of Tweet Type δ_r	δ_{ori}	1.5×10^{-3}	4.8856×10^{-3}
	δ_{ret}	2×10^{-5}	5.1844×10^{-4}
	δ_{quo}	2.5×10^{-6}	5.8166×10^{-35}
	δ_{rply}	5×10^{-6}	1.0185×10^{-52}
Influence Factor between Stances $\gamma_{k',k}$	γ_{ss}	0.9	0.8841
	γ_{sd}	0.1	0.1159
	γ_{ds}	0.5	0.3473
	γ_{dd}	0.5	0.6527
Decay Parameter ω_k	ω_s	3	2.0096
	ω_d	1.5	0.8690

4.5.2 Parameter Estimation on Real Twitter Dataset

In order to trace the influence direction between tweets and users, user relationships with respect to the follower lists of each user were collected to know who will be impacted by each of the prior tweets. Due to the missing information on the data collection on user relationships, we will build the model on the current real Twitter fake news dataset with incomplete user networks. In addition, information collected with tweets will also help in revealing which prior tweet triggers the current one. The posting time of each tweet was converted into a scale from 0 to 6000, which matches the time bounds of the simulated data where 1 day in the real dataset corresponds to 300 time-units in the simulation, and 1 time unit in the simulation equalling 4.8 minutes in the real world.

Table 4.3 shows the estimation result of the parameters from modeling on the real

dataset, which helps us learn how the user stances and tweet types influence the propagation of the fake news story. The base immigrant rates for supporting and denying stances are at the same level but with a higher value on μ_d which shows that original tweets with no obvious supporting stance are slightly more than the original tweets with obvious supporting stance. This indicates that there will be 0.066 supporting immigrants generated every 1 time unit on average, and 0.0995 denying immigrants generated every 1 time unit on average in the simulation, which corresponds to 1 supporting original tweet being generated for every 73 minutes on average, and 1 denying original tweet being generated for every 48 minutes on average in the real-world timeline by converting the time unit in the simulation to the real-world time. The estimation result scale parameter x for the truncated exponential distribution is about 949.5029, which corresponds to the meaning that the average arrival time of immigrants without stance-specified is about 76 hours in the real-world timeline.

The influence factor between stances $\gamma_{k',k}$ shows that a supporting tweet has a 0.7363 probability to trigger a supporting tweet and a 0.2637 probability to trigger a denying stance, while a denying tweet has almost an equal probability ($p = 0.4809$) of generating a denying tweet and a supporting tweet ($p = 0.5191$). This result is similar to what we assumed in the simulation section, which indicates that people tend to believe the fake news story in the current dissemination process of the fake news of covid vaccine will cause female sterilization, as both supporting and denying stances hold a high probability of triggering a supporting tweet generated by a user that connects to the user who generated the prior supporting/denying tweet. γ_{ds} and γ_{dd} are almost the same as our expectation that a denying tweet will have a half chance to generate either a supporting or denying tweet, which matches our observation of the tweets in the dataset where many users keep generating supporting quotes or replies after reviewing the fact-check.

The influence factor of each tweet type δ_r shows a similar trend but slightly deviates from what we expected in the simulation part. The original tweets take the dominant place in affecting users ($\delta_{ori} = 1.6343 \times 10^{-3}$) in terms of generating a tweet related to the fake news story, while retweets hold an even higher influence rate ($\delta_{rply} = 4.0984 \times 10^{-3}$) than the original tweets, by observing multiple retweets from a users' networks and become overwhelmed as the retweets take the smallest cost to be generated than quotes and replies, which correspond to the meaning that 1 tweet will be generated for every 612 users who observe an original tweet on average, and 1 tweet will be generated every 203 users who observe a retweet on average. This also demonstrates the low influence rates for quotes and replies that these two tweet types are generated with a higher cost as it requires comments from users. Replies also require users to click on the tweet and review them, and slide pages for more replies posted in the past such that users may easily lose their patience when replies are generated quickly or with a complex reply chain.

The decay parameters show the overall influence duration of each stance towards the fake news story. The supporting stance μ_s has a shorter impact ($\omega_s = 0.1926$) than the denying stance μ_d ($\omega_d = 0.3932$) which corresponds to the meaning that a supporting tweet will be generated 25 minutes later on average given the influence of a prior tweet, and a denying tweet will be generated 12 minutes later on average given the influence of a prior tweet. This matches the fact that more and more people tend to realize the untrustworthiness of the fake news story. The decay parameter shows that tweets in supporting stance hold a longer influence, which is within our expectation that users tend to believe the fake news instead of the fact-check, or the fact-check articles posted several days later did not attract users' attention anymore. This result reveals the fact that fact-check articles take little effect on combating the fake news in the current fake news dissemination process, and matches the statement

in [Walter and Tukachinsky 2020] that the correction from the fact-check websites cannot entirely revert public opinion to its original status.

Table 4.3: Parameter Estimation: Real Twitter Dataset

Parameter Names		**Estimation**
Immigrant Rate μ_k	μ_s	0.0660
	μ_d	0.0995
TruncExponential Distribution	lower bound	/
	upper bound	/
	scale x	949.5029
Influence Factor of Tweet Type δ_r	δ_{ori}	1.6343×10^{-3}
	δ_{ret}	4.0984×10^{-3}
	δ_{quo}	2.3386×10^{-52}
	δ_{rply}	8.7237×10^{-34}
Influence Factor between Stances $\gamma_{k',k}$	γ_{ss}	0.7363
	γ_{sd}	0.2637
	γ_{ds}	0.5191
	γ_{dd}	0.4809
Decay Parameter ω_k	ω_s	0.1926
	ω_d	0.3932

4.6 Discussion

This paper derives the parameter estimation procedures and expressions for the Multivariate Hawkes Point Processes model for the fake news dissemination process using the Expectation Maximization algorithm applied to the Maximum Likelihood Estimation approach. The parameter estimation performed on the simulated dataset proves the feasibility and the accuracy of the estimation procedures derived through the EM algorithm.

4.6.1 Limitations

From the observation of tweets in the real dataset, there is more than one information source/topic that is relevant to the fake news story during the occurrence of the fake news story which shows the possible interactions between the topics and should be considered in the modeling of the dissemination process as an important component. The incompleteness of the user networks during data collection shows the fact of dynamic user networks over time, which indicates that data collection through Twitter streaming would yield a more complete dataset with real-time user behaviors and networks. The full picture of the fake news dissemination process may reveal a more accurate interaction pattern between users compared to the current findings.

Hashtags and keyword searching should also be considered in the next step of model optimization since they are additional ways of obtaining information aside from being influenced by the information delivered from the user networks. Hashtags provide convenient access for users to learn any information associated with the topic that was generated in the past such that users will receive much more information outside their own user relationships. Keyword searching plays a similar role in the process of information dissemination but does not require any related information from a user's network.

4.6.2 Application

This model explores the influence patterns between user stances during the fake news dissemination process, as well as quantifies the magnitude of the influence of different tweet types, which can be observed and illustrated by the real Twitter dataset reasonably.

Since the model quantifies the influence of any user i received from his/her user network to generate a tweet in type r with stance k, it can be considered as an indicator of the likelihood for user i of believing in the fake news story or not. This likelihood of believing in the misinformation/disinformation can also be studied by learning the users' past online behaviors such as propagating misinformation/disinformation, and following unreliable users in the past, to enhance its accuracy, such that an appropriate pop-up notification can be sent out to the users to notify the possible risk of the unreliable information, and to avoid the possible adverse consequences for users.

Chapter 5

Conclusion and Discussion

Chapter 2, 3, and 4 explores the possible influence the information holds during the dissemination process on online platforms including Yelp and Twitter. Chapter 2 investigated the review influence that holds to the future reviews on the Yelp platform. Chapter 3 and 4 explored the influence of tweet types and the influence between user stances during the process of fake news dissemination on the Twitter platform by simulating the fake Twitter data for model validation and performing Multivariate Hawkes Processes modeling on the real Twitter dataset.

5.1 Research Summary

Based on the analysis from the previous chapters, we can summarize the modeling work and the findings from each topic as follow.

5.1.1 Study of Review Influence on Yelp

In the study of review influence on future reviews on Yelp restaurants, the Lasso Regression model was built on extracted features processed through the Hawkes Point Process model. Specifically, user, review, and restaurants features were extracted from the Yelp dataset provided by Yelp's data challenge in the years 2019 and 2020,

and processed through the Hawkes Point Processes model to capture and aggregate the influence of each feature from prior reviews till the moment before each of the current reviews that were posted, and such feature processing was performed with different values of decay parameter δ setting to the Hawkes Processes to model different influence duration. The processed features with different decay parameters are considered "Hawkes variables" with influences of prior reviews and were implemented in the Lasso Regression model to predict the star rating of each of the current reviews, where the L-1 regularization helps select the significant variables among all similar variables since we have processed each feature with different decay values. The significant variables will present the existence of the aspects where the influence the prior reviews hold on future reviews. Simulation was also performed by generating fake star ratings using the Multinomial Logistic Regression model matching to the shuffled reviews, and re-built the Lasso Regression model on simulated fake reviews to verify the findings from modeling on real Yelp reviews. Another verification step was performed using Logistic Regression building on business-level features such as restaurant stars, number of reviews, and some binary features such as providing lunch or not with a business label of whether the business shows the existence of the review influence.

Both the modeling result on real Yelp data and simulated fake review data suggest that:

- Average star rating has the most significant impact on future reviews with a positive influence on future reviews, which reveals the nature that a business with a higher rating will keep attracting customers to visit and post positive reviews toward the business on Yelp, and thus helps improve rating or at least

remain it unchanged.

- Low star ratings including 1-star, 2-star, and 3-star ratings also hold significant impacts, which indicates that reviews with a lower star rating are more likely to prompt Yelp users who saw the reviews to post new reviews.

- Sentiment-related variables such as sentiment_subjectivity, sentiment_polarity, and the interaction between sentiment_subjectivity and star ratings would trigger new reviews as well since reviews with strong personal feelings or along with extreme star ratings will be more infectious. Reviews with subjective sentiment or positive sentiment are more likely to exert a positive influence on future reviews, in other words, reviews with high star ratings would be triggered.

The final verification step using the Logistic Regression model shows similar findings that:

- Business star rating is significant with a negative coefficient showing that it is more likely that past reviews of a business with a lower average star rating will influence the future of the current business. Furthermore, a lower average star rating is caused by accumulative reviews with a low star rating which verify the finding from the Lasso Regression modeling that 1-star, 2-star, and 3-star ratings are the variables that have significant influences on future reviews.

- The number of reviews a business has is significant which indicates that a business attracts relatively more customers to visit and post reviews will hold an influence on future reviews.

- The opening years of the business is significant however with a negative coefficient, from which we could infer that businesses with a relatively long-term

operation have more reviews posted at the early stage of Yelp's development with less influence due to the limited number of users.

5.1.2 Influence of User Stances and Tweet Types on Fake News Dissemination Process on Twitter

The study of the influence of user stances and tweet types on the Fake News Dissemination Process on Twitter is performed by simulation and modeling of the fake news dissemination process on Twitter using Multivariate Hawkes Processes. We first developed a new simulation method based on a Multivariate Hawkes Processes model extended and adapted to the fake news dissemination process on Twitter, where it considered the user networks, the influence of user stances, and the influence of tweet types. Maximum Likelihood Estimation with the Expectation Maximization algorithm was derived for the Multivariate Hawkes Process model, and was tested by estimating the parameter values of the simulated dataset generated through the proposed simulation approach with predetermined parameter values. After the verification of the parameter estimation, the model was built on the real Twitter dataset collected through Twitter API on one fake news story that occurred on Twitter, and the verified parameter estimation procedures were performed to estimate the parameter values of the real dataset, which revealed how stances toward the fake news between users and tweet types would influence the fake news dissemination process.

The final estimation result suggests that:

- Immigrants for supporting stance μ_s and denying stance μ_d have similar rates but denying rate is a little bit higher which corresponds to a little bit more original tweets in denying stance in the fake news dataset we collected from

Twitter.

- The influence factor between stances $\gamma_{k',k}$ indicates that people tend to believe in the fake news story in the current dissemination process of the fake news of covid vaccine will cause female sterilization, as both supporting and denying stances hold a high probability of triggering a supporting tweet generated by a user that connects to the user who generated the prior supporting/denying tweet.

- The influence factor of each tweet type δ_r shows that the original tweets and retweets take the dominant place in affecting users in terms of generating a tweet related to the fake news story. Quotes and replies hold a similar influencing ability which is relatively lower than the original tweets and retweets hold which can be explained as the quotes are seldomly generated by the users, and replies require extra behaviors and costs for reading.

- The decay parameter ω_k shows the same pattern as we expected that denying tweets hold a longer influence than the supporting tweets do in the fake news dataset we collected, but both of them produce long-term influences than our expectation in the simulation.

5.2 Limitations and Future Work

There are some limitations for each of the topics that we can address as indicators for the applicable future work.

For the analysis of review influence on future reviews, due to the fact that the businesses in the public Yelp dataset were partially selected such that most are located in

Las Vegas and Phoenix, which may cause bias on the result and prevent the conclusions to be generalized to businesses elsewhere. In addition, the results were obtained from the dataset in years 2019 and 2020, whereas the dataset in most recent years may produce different conclusions. We can improve the research in multiple directions: apply a larger and more general business dataset; increase the number of simulations; particularly, further analysis could be performed to explore specific relationships between prior reviews and future reviews with respect to different aspects (e.g. how a review feature would affect other features specifically).

For the study of the fake news dissemination process on Twitter, there are several limitations found through the study that can be considered to be further improved in the next step.

- Online users may follow/un-follow other users during the fake news dissemination process, such that will cause a dynamic user network and thus impact who will be influenced by the information before and after the network change. Thus, real-time data collection on both tweets and user information is required in the future to further improve the model's accuracy.

- Text factors such as tweet sentiment can be considered an influential factor in the model since quotes (tweets with comments) and replies will provide additional information from the users that may attract more users to respond toward both the fake news and the comments.

- It is possible that additional topics related to the ongoing fake news story on the platform emerge on the online platform, which indicates that more than one process exists simultaneously and interacts with each other due to the information-sharing behaviors of the users. Such interactions within the rele-

vant online topics should also be considered to quantify the influence that the current process may receive from other processes.

- Hashtag is another factor that may influence the tweet explosion since hashtag provides the function of accessing all tweets that contain the hashtag such that users clicking the hashtag are able to review all prior tweets and will possibly be influenced by them.

- More stances can be considered aside from the denying (not supporting) stances such as questioning and commenting stances to improve the model accuracy. This improvement option should be performed when the size of the dataset is large enough for any stance of tweets to be accurately estimated on the parameters associated with it.

The application of this study can be discussed through two parts: simulation and modeling. Although the proposed simulation approach is developed based on the cluster-based and intensity-based simulation methods in Hawkes Point Process theory, the simulated dataset itself can be applied in the testing for modeling of the fake news dissemination process using other models or methods. The modeling work in this study can be extended to a long-term learning process of users' past behaviors in spreading misinformation/disinformation, such that this model can be used in calculating the intensity of the misinformation/disinformation the users may receive according to their user networks, and make appropriate reactions to the users to notify them of the possible risk that the information would cause, hence prevent the negative effect from happening for both users and the online platforms.

Bibliography

Erdős, Paul, Alfréd Rényi, et al. (1960). "On the evolution of random graphs". In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1, pp. 17–60.

Hawkes, Alan G (1971). "Spectra of some self-exciting and mutually exciting point processes". In: *Biometrika* 58.1, pp. 83–90.

Hawkes, Alan G and David Oakes (1974). "A cluster process representation of a self-exciting process". In: *Journal of Applied Probability* 11.3, pp. 493–503.

Dempster, Arthur P, Nan M Laird, and Donald B Rubin (1977). "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1, pp. 1–22.

Ogata, Yosihiko (1981). "On Lewis' simulation method for point processes". In: *IEEE transactions on information theory* 27.1, pp. 23–31.

Schoenberg, Isaac J (1988). "Contributions to the problem of approximation of equidistant data by analytic functions". In: *IJ Schoenberg Selected Papers*. Springer, pp. 3–57.

Freed, Andrew M (2005). "Earthquake triggering by static, dynamic, and postseismic stress transfer". In: *Annual Review of Earth and Planetary Sciences* 33.1, pp. 335–367.

Luke, Sean et al. (2005). "Mason: A multiagent simulation environment". In: *Simulation* 81.7, pp. 517–527.

Møller, Jesper and Jakob G Rasmussen (2005). "Perfect simulation of Hawkes processes". In: *Advances in applied probability* 37.3, pp. 629–646.

— (2006). “Approximate simulation of Hawkes processes”. In: *Methodology and Computing in Applied Probability* 8.1, pp. 53–64.

Bowsher, Clive G (2007). “Modelling security market events in continuous time: Intensity based, multivariate point process models”. In: *Journal of Econometrics* 141.2, pp. 876–912.

Hagberg, Aric, Pieter Swart, and Daniel S Chult (2008). *Exploring network structure, dynamics, and function using NetworkX*. Tech. rep. Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Hu, Nan, Ling Liu, and Jie Jennifer Zhang (2008). “Do online reviews affect product sales? The role of reviewer characteristics and temporal effects”. In: *Information Technology and management* 9.3, pp. 201–214.

Li, Xiaochen et al. (2008). “Agent-based social simulation and modeling in social computing”. In: *International Conference on Intelligence and Security Informatics*. Springer, pp. 401–412.

Liu, Dechun and Xi Chen (2011). “Rumor propagation in online social networks like twitter–a simulation study”. In: *2011 Third International Conference on Multimedia Information Networking and Security*. IEEE, pp. 278–282.

Lewandowsky, Stephan et al. (2012). “Misinformation and its correction: Continued influence and successful debiasing”. In: *Psychological science in the public interest* 13.3, pp. 106–131.

Chen, Wei, Laks VS Lakshmanan, and Carlos Castillo (2013). “Information and influence propagation in social networks”. In: *Synthesis Lectures on Data Management* 5.4, pp. 1–177.

Dassios, Angelos and Hongbiao Zhao (2013). “Exact simulation of Hawkes process with exponentially decaying intensity”. In: *Electronic Communications in Probability* 18, pp. 1–13.

Lu, Yuqing et al. (2013). "Simultaneously detecting fake reviews and review spammers using factor graph model". In: *Proceedings of the 5th annual ACM web science conference*, pp. 225–233.

Mukherjee, Arjun et al. (2013). "What yelp fake review filter might be doing?" In: *Proceedings of the international AAAI conference on web and social media*. Vol. 7. 1.

Weng, Lilian, Filippo Menczer, and Yong-Yeol Ahn (2013). "Virality prediction and community structure in social networks". In: *Scientific reports* 3.1, pp. 1–6.

Ahmed, Lubaid and Abdolreza Abhari (2014). "Agent-based simulation of twitter for building effective recommender system". In: *Proceedings of the 17th Communications & Networking Simulation Symposium*, pp. 1–7.

Ngo-Ye, Thomas L and Atish P Sinha (2014). "The influence of reviewer engagement characteristics on online review helpfulness: A text regression model". In: *Decision Support Systems* 61, pp. 47–58.

Parikh, Anish et al. (2014). "Motives for reading and articulating user-generated restaurant reviews on Yelp. com". In: *Journal of Hospitality and Tourism Technology*.

Boshmaf, Yazan et al. (2015). "Integro: leveraging victim prediction for robust fake account detection in osns." In: *NDSS*. Vol. 15, pp. 8–11.

Pinto, Julio Cesar Louzada and Tijani Chahed (2015). "Modeling user and topic interactions in social networks using Hawkes processes". In: *EAI Endorsed Transactions on Cloud Systems* 1.3.

Pinto, Julio Cesar Louzada, Tijani Chahed, and Eitan Altman (2015). "Trend detection in social networks using Hawkes processes". In: *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pp. 1441–1448.

Xu, Yun, Xinhui Wu, and Qinxia Wang (2015). "Sentiment Analysis of Yelp's Ratings Based on Text Reviews". In: *Stanford University* 17, pp. 117–120.

Asghar, Nabiha (2016). "Yelp dataset challenge: Review rating prediction". In: *arXiv preprint arXiv:1605.05362*.

Kc, Santosh and Arjun Mukherjee (2016). "On the temporal dynamics of opinion spamming: Case studies on yelp". In: *Proceedings of the 25th International Conference on World Wide Web*, pp. 369–379.

Kobayashi, Ryota and Renaud Lambiotte (2016). "Tideh: Time-dependent hawkes process for predicting retweet dynamics". In: *Tenth International AAAI Conference on Web and Social Media*.

Law, Baron and Frederi Viens (2016). "Hawkes Processes and Their Applications to High-Frequency Data Modeling". In: *Handbook of High-Frequency Trading and Modeling in Finance*, pp. 183–219.

Luca, Michael (2016). "Reviews, reputation, and revenue: The case of Yelp. com". In: *Com (March 15, 2016). Harvard Business School NOM Unit Working Paper* 12-016.

Lukasik, Michal et al. (2016). "Hawkes processes for continuous time sequence classification: an application to rumour stance classification in twitter". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 393–398.

Mishra, Swapnil, Marian-Andrei Rizoiu, and Lexing Xie (2016). "Feature driven and point process approaches for popularity prediction". In: *Proceedings of the 25th ACM international on conference on information and knowledge management*, pp. 1069–1078.

Network, The International Fact-Checking (Nov. 2016). *An open letter to Mark Zuckerberg from the world's fact-checkers*. URL: https://www.poynter.org/

fact-checking/2016/an-open-letter-to-mark-zuckerberg-from-the-worlds-fact-checkers/.

Pinto, Julio Cesar Louzada (2016). "Information diffusion and opinion dynamics in social networks". PhD thesis. Institut National des Télécommunications.

Sakas, Damianos P and Apostolos S Sarlis (2016). "Library promotion methods and tools modeling and simulation on Twitter". In: *Library Review*.

Serrano, Emilio and Carlos A Iglesias (2016). "Validating viral marketing strategies in Twitter via agent-based social simulation". In: *Expert Systems with Applications* 50, pp. 140–150.

Simon, Graham (2016). "Hawkes processes in finance: A review with simulations". In.

Wang, Yichen et al. (2016). "Coevolutionary latent feature processes for continuous-time user-item interactions". In: *Advances in neural information processing systems* 29.

Zipkin, Joseph R et al. (2016). "Point-process models of social network interactions: Parameter estimation and missing data recovery". In: *European journal of applied mathematics* 27.3, pp. 502–529.

Allcott, Hunt and Matthew Gentzkow (2017). "Social media and fake news in the 2016 election". In: *Journal of economic perspectives* 31.2, pp. 211–36.

Bruns, Axel (2017). "Echo chamber? What echo chamber? Reviewing the evidence". In: *6th Biennial Future of Journalism Conference (FOJ17)*.

Burkhardt, Joanna M (2017). "History of fake news". In: *Library Technology Reports* 53.8, pp. 5–9.

Cao, Qi et al. (2017). "Deephawkes: Bridging the gap between prediction and understanding of information cascades". In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1149–1158.

Farajtabar, Mehrdad et al. (2017). "Fake news mitigation via point process based intervention". In: *International conference on machine learning*. PMLR, pp. 1097–1106.

Granik, Mykhailo and Volodymyr Mesyura (2017). "Fake news detection using naive Bayes classifier". In: *2017 IEEE first Ukraine conference on electrical and computer engineering (UKRCON)*. IEEE, pp. 900–903.

Huang, Wei-Bo et al. (2017). "The application of big data on we-media information and crisis management system". In: *2017 4th International Conference on Systems and Informatics (ICSAI)*. IEEE, pp. 1579–1584.

Kirchner, Matthias (2017). "An estimation procedure for the Hawkes process". In: *Quantitative Finance* 17.4, pp. 571–595.

Li, Sha et al. (2017). "Fm-hawkes: A hawkes process based approach for modeling online activity correlations". In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1119–1128.

Morse, Steven T (2017). "Persistent cascades and the structure of influence in a communication network". PhD thesis. Massachusetts Institute of Technology.

Porter, Michael (2017). "Multivariate hawkes point process models for social systems". In: *Proceedings of the 62nd World Statistics Congress of the International Statistical Institute*.

Rahimi, Sohrab, Andris Clio, and Liu. Xi (2017). "Using yelp to find romance in the city: A case of restaurants in four cities". In: *Proceedings of the 3rd ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics*, pp. 1–8.

Rizoiu, Marian-Andrei et al. (2017). "A tutorial on hawkes processes for events in social media". In: *arXiv preprint arXiv:1708.06401*.

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

Shu, Kai, Suhang Wang, and Huan Liu (2017). "Exploiting tri-relationship for fake news detection". In: *arXiv preprint arXiv:1712.07709* 8.

Yang, Sung-Byung et al. (2017). "An empirical examination of online restaurant reviews on Yelp. com: A dual coding theory perspective". In: *International Journal of Contemporary Hospitality Management*.

Zhou, Shasha and Bin Guo (2017). "The order effect on online review helpfulness: A social influence perspective". In: *Decision Support Systems* 93, pp. 77–87.

Dungs, Sebastian et al. (2018). "Can rumour stance alone predict veracity?" In: *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 3360–3370.

El Maazouz, Yassine and Mohammed Amine Bennouna (2018). "Simulating rare events: Hawkes process applied to Twitter". PhD thesis. Ecole polytechnique X.

Helmstetter, Stefan and Heiko Paulheim (2018). "Weakly supervised learning for fake news detection on Twitter". In: *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, pp. 274–277.

Lazer, David MJ et al. (2018). "The science of fake news". In: *Science* 359.6380, pp. 1094–1096.

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

Pentina, Iryna, Ainsworth Anthony Bailey, and Lixuan Zhang (2018). "Exploring effects of source similarity, message valence, and receiver regulatory focus on yelp review persuasiveness and purchase intentions". In: *Journal of Marketing Communications* 24.2, pp. 125–145.

Rapp, David N and Nikita A Salovich (2018). "Can't we just disregard fake news? The consequences of exposure to inaccurate information". In: *Policy Insights from the Behavioral and Brain Sciences* 5.2, pp. 232–239.

Shao, Chengcheng, Giovanni Luca Ciampaglia, et al. (2018). "The spread of low-credibility content by social bots". In: *Nature communications* 9.1, pp. 1–9.

Shao, Chengcheng, Pik-Mai Hui, et al. (2018). "Anatomy of an online misinformation network". In: *Plos one* 13.4, e0196087.

Tandoc Jr, Edson C, Zheng Wei Lim, and Richard Ling (2018). "Defining "fake news" A typology of scholarly definitions". In: *Digital journalism* 6.2, pp. 137–153.

Törnberg, Petter (2018). "Echo chambers and viral misinformation: Modeling fake news as complex contagion". In: *PLoS one* 13.9, e0203958.

Alvari, Hamidreza and Paulo Shakarian (2019). "Hawkes process for understanding the influence of pathogenic social media accounts". In: *2019 2nd International Conference on Data Intelligence and Security (ICDIS)*. IEEE, pp. 36–42.

Aono, Tavanleuang VANTA Masaki (2019). "Fake review detection focusing on emotional expressions and extreme rating". In.

Beskow, David M and Kathleen M Carley (2019). "Agent based simulation of bot disinformation maneuvers in Twitter". In: *2019 Winter simulation conference (WSC)*. IEEE, pp. 750–761.

Bovet, Alexandre and Hernán A Makse (2019). "Influence of fake news in Twitter during the 2016 US presidential election". In: *Nature communications* 10.1, pp. 1–14.

Islam, Mohammad Raihanul, Sathappan Muthiah, and Naren Ramakrishnan (2019). "RumorSleuth: joint detection of rumor veracity and user stance". In: *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, pp. 131–136.

Li, Chenlong, Zhanjie Song, and Xu Wang (2019). "Nonparametric method for modeling clustering phenomena in emergency calls under spatial-temporal self-exciting point processes". In: *IEEE Access* 7, pp. 24865–24876.

Sihombing, Andre and Alvis Cheuk Ming Fong (2019). "Fake review detection on yelp dataset using classification techniques in machine learning". In: *2019 International Conference on contemporary Computing and Informatics (IC3I)*. IEEE, pp. 64–68.

Vinson, David W, Rick Dale, and Michael N Jones (2019). "Decision contamination in the wild: Sequential dependencies in online review ratings". In: *Behavior research methods* 51.4, pp. 1477–1484.

Zhou, Kaimin et al. (2019). "Early rumour detection". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 1614–1623.

Zimmer, Franziska et al. (2019). "Fake news in social media: Bad algorithms or biased users?" In: *Journal of Information Science Theory and Practice* 7.2, pp. 40–53.

Dutta, Hridoy Sankar et al. (2020). "HawkesEye: Detecting fake retweeters using Hawkes process and topic modeling". In: *IEEE Transactions on Information Forensics and Security* 15, pp. 2667–2678.

Goindani, Mahak and Jennifer Neville (2020). "Social reinforcement learning to combat fake news spread". In: *Uncertainty in Artificial Intelligence*. PMLR, pp. 1006–1016.

Kong, Quyu, Marian-Andrei Rizoiu, and Lexing Xie (2020). "Modeling information cascades with self-exciting processes via generalized epidemic models". In: *proceedings of the 13th international conference on web search and data mining*, pp. 286–294.

Nie, H Ruda et al. (2020). "Modelling user influence and rumor propagation on Twitter using Hawkes processes". In: *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, pp. 637–656.

O'Rourke, Ciara (Dec. 2020). *No, Pfizer's head of research didn't say the COVID-19 vaccine will make women infertile*. URL: https://www.politifact.com/factchecks/2020/dec/10/blog-posting/no-pfizers-head-research-didnt-say-covid-19-vaccin/.

Shrivastava, Gulshan et al. (2020). "Defensive modeling of fake news through online social networks". In: *IEEE Transactions on Computational Social Systems* 7.5, pp. 1159–1167.

Tian, Lin, Xiuzhen Zhang, and Min Peng (2020). "FakeFinder: twitter fake news detection on mobile". In: *Companion Proceedings of the Web Conference 2020*, pp. 79–80.

Tsfati, Yariv et al. (2020). "Causes and consequences of mainstream media dissemination of fake news: literature review and synthesis". In: *Annals of the International Communication Association* 44.2, pp. 157–173.

Walter, Nathan and Riva Tukachinsky (2020). "A meta-analytic examination of the continued influence of misinformation in the face of correction: How powerful is it, why does it happen, and how to stop it?" In: *Communication research* 47.2, pp. 155–177.

Wang, Yufang et al. (2020). "Regional influenza prediction with sampling twitter data and PDE model". In: *International journal of environmental research and public health* 17.3, p. 678.

Yelp (Feb. 2020). *Yelp open dataset*. URL: https://www.yelp.com/dataset/.

Zhao, Zilong et al. (2020). "Fake news propagates differently from real news even at early stages of spreading". In: *EPJ data science* 9.1, p. 7.

Murayama, Taichi et al. (2021). "Modeling the spread of fake news on Twitter". In: *Plos one* 16.4, e0250419.

Qu, Ao and Ismael Lemhadri (2021). "A Graph Approach to Simulate Twitter Activities with Hawkes Processes". In: *2021 4th International Conference on Mathematics and Statistics*, pp. 80–85.

Sano, Yukie et al. (2021). "Simulation of Information Spreading on Twitter Concerning Radiation After the Fukushima Nuclear Power Plant Accident". In: *Frontiers in Physics* 9, p. 357.

Textblob (Jan. 2021). *TextBlob: Simplified Text Preprocessing*. URL: https://textblob.readthedocs.io/en/dev/.

Aslam, Salman (Feb. 2022). *Twitter by the numbers (2022): Stats, Demographics & Fun Facts*. URL: https://www.omnicoreagency.com/twitter-statistics/.

Davoudi, Mansour, Mohammad R Moosavi, and Mohammad Hadi Sadreddini (2022). "DSS: A hybrid deep model for fake news detection using propagation tree and stance network". In: *Expert Systems with Applications* 198, p. 116635.

Jiang, Yichen (2022). *Github-Twitter-FakeNews-MHP*. URL: https://github.com/yjiang-github/Twitter-FakeNews-MHP.

Shlomovich, Leigh et al. (2022). "Parameter Estimation of Binned Hawkes Processes". In: *Journal of Computational and Graphical Statistics*, pp. 1–11.

Tondulkar, Rohan et al. (2022). "Hawkes Process Classification through Discriminative Modeling of Text". In: *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–8.

Appendices

Appendix A

B-Spline Basis Function

The basic framework of B-Spline curve has been created by Schoenberg on 1946 [Schoenberg 1988], and has been developed to adjust different application such as modeling of 3-D geometry shape or interpolation of fluctuating data points for smoothing purpose. A k-order B-Spline curve is composed by a set of linear-combined control points P_i and B-Spline basis functions denoted as $N_{i,k}(t)$, and each control point is associated with a basis function in a recurrence relation such that:

$$N_{i,k}(t) = N_{i,k-1}(t)\frac{t - t_i}{t_{i+k-1} - t_i} + N_{i+1,k-1}(t)\frac{t_{i+k} - t}{t_{i+k} - t_{i+1}} \tag{A.1}$$

$$N_{i,1} = \begin{cases} 1 & if \quad t_i \leq t \leq t_{i+1} \\ 0 & otherwise \end{cases}$$

The shape of the B-Spline basis function is determined by the knot vector:

$$T = (t_0, t_1, ..., t_{k-1}, t_k, t_{k+1}, ..., t_{n-1}, t_n, t_{n+1}, ..., t_{n+k}) \tag{A.2}$$

The number of elements of the knot vector is defined by the sum of the number of control points and the order of the B-Spline curve $(n + k + 1)$.

Appendix B

Expectation Maximization for Multivariate Hawkes Processes

B.1 Log-Likelihood Function

The detailed derivation of the Maximum Likelihood function shows as follows:

$$\log L_k(t;\theta) = -\int_0^T \lambda_k(t;\theta)\mathrm{d}t + \sum_{j=1}^{N(t)} \log \lambda_k(t;\theta) \tag{B.1}$$

$$= -\int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t-t_j;\theta)\right)\mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k(t) + \sum_{l=1}^{N(t_j)} h_k(t_j - t_l;\theta)\right) \tag{B.2}$$

$$\begin{aligned} = &-\int_0^T \left(\mu_k \frac{xe^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)} \delta_{r_j}\gamma_{k_j,k} n_j g_k(t-t_j)\right)\mathrm{d}t \\ &+ \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{xe^{-xt}}{1-e^{-xT}} + \sum_{l=1}^{N(t_j)} \delta_{r_l}\gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right) \end{aligned} \tag{B.3}$$

$$\begin{aligned} = &-\left(\frac{\mu_k}{1-e^{-xT}} \cdot \int_0^T xe^{-xt}\mathrm{d}t\right) - \sum_{j=1}^{N(t)} \int_{t_j}^T \delta_{r_j}\gamma_{k_j,k} n_j g_k(t-t_j)\mathrm{d}t \\ &+ \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{xe^{-xt}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}\gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right) \end{aligned} \tag{B.4}$$

$$= -\left(\frac{\mu_k}{1-e^{-xT}} \cdot (1-e^{-xt}\Big|_0^T)\right) - \sum_{j=1}^{N(t)} \int_{t_j}^{T} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t-t_j) \mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{x e^{-xt}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right) \tag{B.5}$$

$$= -\left(\frac{\mu_k}{1-e^{-xT}} \cdot (1-e^{-xT})\right) - \sum_{j=1}^{N(t)} \int_{t_j}^{T} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t-t_j) \mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{x e^{-xt}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right) \tag{B.6}$$

$$= -\mu_k - \sum_{j=1}^{N(t)} \int_{t_j}^{T} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t-t_j) \mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{x e^{-xt}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)\right) \tag{B.7}$$

$$\tag{B.8}$$

This process can be considered as a Multivariate Hawkes Process such that the log-likelihood function can be expressed as the sum of the individual log-likelihoods:

$$\log L(t;\theta) = \sum_{k \in K} \log L_k(t;\theta) \tag{B.9}$$

$$= \sum_{k \in K} \left[-\int_0^T \lambda_k(t;\theta) + \sum_{j=1}^{N(t)} \log \lambda_k(t_j;\theta) \right] \tag{B.10}$$

$$= \sum_{k \in K} \left[-\int_0^T \left(\mu_k \frac{x e^{-xt}}{1-e^{-xT}} + \sum_{j=1}^{N(t)} h_k(t-t_j;\theta)\right) \mathrm{d}t + \sum_{j=1}^{N(t)} \log\left(\mu_k \frac{x e^{-xt}}{1-e^{-xT}} + \sum_{l=1}^{N(t_j)} h_k(t_j - t_l;\theta)\right) \right] \tag{B.11}$$

$$
\begin{aligned}
&= \sum_{k \in K} \Bigg[- \mu_k - \sum_{j=1}^{N(t)} \int_{t_j}^{T} \delta_{r_j} \gamma_{k_j,k} n_j g_k(t - t_j) \mathrm{d}t \\
&\quad + \sum_{j=1}^{N(t)} \log \left(\mu_k \frac{x e^{-xt}}{1 - e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l) \right) \Bigg]
\end{aligned} \tag{B.12}
$$

We will convert the log-likelihood function to the Q function, which is the expectation of the log-likelihood function, and the lower bound of the log-likelihood function as well. Based on Jensen's inequality, we have:

$$
\log L_k(t; \theta) = \sum_{j=1}^{N(t)} \log \left(\mu_k(t) + \sum_{l=1}^{N(t_j)} h_k(t_j - t_l; \theta) \right) - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j; \theta) \right) \mathrm{d}t \tag{B.13}
$$

$$
\begin{aligned}
&= \sum_{j=1}^{N(t)} \log \left(p_{jj}^k \cdot \frac{\mu_k(t)}{p_{jj}^k} + \sum_{l=1}^{N(t_j)} p_{jl}^k \cdot \frac{h_k(t_j - t_l; \theta)}{p_{jl}^k} \right) \\
&\quad - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j; \theta) \right) \mathrm{d}t
\end{aligned} \tag{B.14}
$$

$$
\begin{aligned}
&\geq \sum_{j=1}^{N(t)} \left[p_{jj}^k \log \left(\frac{\mu_k(t)}{p_{jj}^k} \right) + \sum_{l=1}^{N(t_j)} p_{jl}^k \log \left(\frac{h_k(t_j - t_l; \theta)}{p_{jl}^k} \right) \right] \\
&\quad - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j; \theta) \right) \mathrm{d}t
\end{aligned} \tag{B.15}
$$

$$
\begin{aligned}
&= \sum_{j=1}^{N(t)} \left[p_{jj}^k \log \mu_k(t) - p_{jj}^k \log p_{jj}^k + \sum_{l=1}^{N(t_j)} \left(p_{jl}^k \log h_k(t_j - t_l; \theta) - p_{jl}^k \log p_{jl} \right) \right] \\
&\quad - \int_0^T \left(\mu_k(t) + \sum_{j=1}^{N(t)} h_k(t - t_j; \theta) \right) \mathrm{d}t
\end{aligned} \tag{B.16}
$$

$$
\begin{aligned}
&= \sum_{j=1}^{N(t)} p_{jj}^{k} \log \mu_k(t) + \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) - \sum_{j=1}^{N(t)} p_{jj}^{k} \log p_{jj}^{k} \\
&\quad - \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log p_{jl}^{k} - \mu_k - \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t
\end{aligned} \tag{B.17}
$$

$$
\begin{aligned}
&= \sum_{j=1}^{N(t)} p_{jj}^{k} \log(\mu_k \frac{x e^{-xt}}{1 - e^{-xT}}) - \mu_k + \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) \\
&\quad - \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t + C
\end{aligned} \tag{B.18}
$$

Therefore, the lower bound of the log-likelihood function $Q_k(T; \theta)$ is expressed as:

$$
\begin{aligned}
Q_k(T; \theta) = &\sum_{j=1}^{N(t)} p_{jj}^{k} \log(\mu_k \frac{x e^{-xt}}{1 - e^{-xT}}) - \mu_k + \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) \\
&- \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t + C
\end{aligned} \tag{B.19}
$$

and the overall Q function shows as follows:

$$
\begin{aligned}
Q(T; \theta) = &\sum_{k \in K} \sum_{j=1}^{N(t)} p_{jj}^{k} \log(\mu_k \frac{x e^{-xt}}{1 - e^{-xT}}) - \sum_{k \in K} \mu_k + \sum_{k \in K} \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k} \log h_k(t_j - t_l; \theta) \\
&- \sum_{k \in K} \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j; \theta) \mathrm{d}t + C
\end{aligned} \tag{B.20}
$$

B.2 Expectation Step

For E-step at $s+1$ iteration, the probability distribution that any event Z_j in stance k is an immigrant event is calculated as:

$$p_{jj}^{k\ (s+1)} = \frac{\mu_k^{(s)}(t_j)}{\lambda_k(t_j;\theta^{(s)})} = \frac{\mu_k^{(s)}\frac{xe^{-xt_j}}{1-e^{-xT}}}{\mu_k^{(s)}\frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} h_k(t_j - t_l;\theta^{(s)})} \tag{B.21}$$

$$= \frac{\mu_k^{(s)}\frac{xe^{-xt_j}}{1-e^{xT}}}{\mu_k^{(s)}\frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)}\gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)} \tag{B.22}$$

The probability that any event Z_j is a descendant influenced by prior event l is calculated as:

$$p_{jl}^{k\ (s+1)} = \frac{h_k(t_j;\theta)^{(s)}}{\lambda_k(t_j;\theta^{(s)})} = \frac{h_k(t_j - t_l;\theta^{(s)})}{\mu_k^{(s)}\frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} h_k(t_j - t_l;\theta^{(s)})} \tag{B.23}$$

$$= \frac{\delta_{r_l}^{(s)}\gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)}{\mu_k^{(s)}\frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)}\gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)} \tag{B.24}$$

B.3 Maximization Step

At each iteration of the EM algorithm, M-step will maximize the lower bound by maximizing the values of the parameters such that the algorithm iterates over the E-step and M-step and keeps searching for the optimal solutions until the Q function converges.

The maximum value of parameter θ occurs when $\frac{\partial Q(T;\theta)}{\partial \theta} = 0$ such that for μ_k in $Q_k(T;\theta)$ we have:

$$\frac{\partial Q_k(T;\theta)}{\partial \mu_k} = \sum_{j=1}^{N(t)} p_{jj}^k \frac{1}{\mu_k \cdot \frac{xe^{-xt}}{1-e^{-xT}}} \cdot \frac{xe^{-xt}}{1-e^{-xT}} - 1 = 0 \tag{B.25}$$

$$\frac{1}{\mu_k}\sum_{j=1}^{N(T)} p_{jj}^k = 1 \tag{B.26}$$

$$\mu_k = \sum_{j=1}^{N(t)} p_{jj}^k \tag{B.27}$$

Particularly, as the simulation introduced in the section 3.4.4, the number of immigrants is modeled and simulated based on the Poisson distribution (Pois) with a mean of overall immigrant rate which equals to $\mu_k T$, such that the equation of updating μ_k at each iteration $s+1$ of calculation is equivalent to:

$$\mu_k^{(s+1)} = \frac{\sum_{j=1}^{N(t)} p_{jj}^{k\ (s)}}{T} \tag{B.28}$$

For the other parameters associated with the self-exciting process when applying the exponential kernel function, we have:

$$\frac{\partial\left(\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^k \log h_k(t_j - t_l;\theta)\right)}{\partial\theta} - \frac{\partial\left(\sum_{j=1}^{N(t)}\int_{t_j}^{T} h_k(t-t_j;\theta)\mathrm{d}t\right)}{\partial\theta} = 0 \tag{B.29}$$

Therefore:

$$\frac{\partial\left(\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^k \log h_k(t_j - t_l;\theta)\right)}{\partial\theta} = \frac{\partial\left(\sum_{j=1}^{N(t)}\int_{t_j}^{T} h_k(t-t_j;\theta)\mathrm{d}t\right)}{\partial\theta} \tag{B.30}$$

$$\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\theta h_k(t_j - t_l;\theta)}{h_k(t_j - t_l;\theta)} = \sum_{j=1}^{N(t)} \left(\partial_\theta H_k(T-t_j;\theta) - \partial_\theta H_k(0;\theta)\right) \tag{B.31}$$

$$\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\theta(\delta_{r_l}\gamma_{k_l,k_j} n_l g_k(t_j - t_l))}{\delta_{r_l}\gamma_{k_l,k_j} n_l g_k(t_j - t_l)} = \sum_{j=1}^{N(t)} \partial_\theta\left(\delta_{r_j}\gamma_{k_j,k_j} n_j (G_k(T-t_j) - G_k(0))\right) \tag{B.32}$$

For exponential kernel $g_k(t-t_j) = \omega_k e^{-\omega_k(t-t_j)}$, we have:

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\theta(\delta_{r_l}\gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l}\gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{j=1}^{N(t)} \partial_\theta \left(\delta_{r_j}\gamma_{k_j,k} n_j \cdot (-e^{-\omega_k(T-t_j)} - e^{-\omega_k(0)})\right) \tag{B.33}$$

$$= \sum_{j=1}^{N(t)} \partial_\theta \left(\delta_{r_j}\gamma_{k_j,k} n_j \cdot (1 - e^{-\omega_k(T-t_j)})\right) \tag{B.34}$$

where ω_k is the decay parameter that controls the decay speed of the process regarding stance k.

For parameterγ, we have:

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\gamma(\delta_{r_l}\gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l}\gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{j=1}^{N(t)} \partial_\gamma \left(\delta_{r_j}\gamma_{k_j,k} n_j \cdot (1 - e^{-\omega_k(T-t_j)})\right) \tag{B.35}$$

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\delta_{r_l} n_l \omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l}\gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{j=1}^{N(t)} \delta_{r_j} n_j \left(1 - e^{-\omega_k(T-t_j)})\right) \tag{B.36}$$

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{1}{\gamma_{k_l,k_j}} = \sum_{j=1}^{N(t)} \delta_{r_j} n_j \left(1 - e^{-\omega_k(T-t_j)})\right) \tag{B.37}$$

Since γ_{k_l,k_j} relates to the stance of lth tweet Z_l, for each specific stance k' (denying, supporting), we have:

$$\sum_{j=1}^{N(t)} \sum_{l:k_l=k'}^{N(t_j)} p_{jl}^k \frac{1}{\gamma_{k',k}} = \sum_{j=1}^{N(t)} \delta_{r_j} n_j \left(1 - e^{-\omega_k(T-t_j)})\right) \tag{B.38}$$

$$\frac{1}{\gamma_{k',k}} \sum_{j=1}^{N(t)} \sum_{l:k_l=k'}^{N(t_j)} p_{jl}^k = \sum_{j=1}^{N(t)} \delta_{r_j} n_j \left(1 - e^{-\omega_k(T-t_j)})\right) \tag{B.39}$$

$$\gamma_{k',k} = \frac{\sum_{j=1}^{N(t)} \sum_{l:k_l=k'}^{N(t_j)} p_{jl}^k}{\sum_{j=1}^{N(t)} \delta_{r_j} n_j \left(1 - e^{-\omega_k(T-t_j)})\right)} \tag{B.40}$$

Consider the case that a fake news story outbreaks on Twitter initially, diminishes over time, and vanishes eventually, the occurring time of each event t_j should deviate from the total time T of the observation as the intensity decays over time, such that $e^{-\omega_k(T-t_j)} \approx 0$. Thus, the above expression can be simplified to its closed-form expression:

$$\gamma_{k',k} \approx \frac{\sum_{j=1}^{N(t)} \sum_{l:k_l=k'}^{N(t_j)} p_{jl}^k}{\sum_{j=1}^{N(t)} \delta_{r_j} n_j} \tag{B.41}$$

Particularly, as the **Assumption 2** we made in section 3.3.2, all the retweets should hold the same stance as the tweet triggers it, such that the between-stance factor $\gamma_{k',k}$ should take effect on generating quotes and replies with the following adjustment on the expression at each iteration $s+1$:

$$\gamma_{k',k}{}^{(s+1)} = \frac{\sum_{j:r_j \in \{quo,rply\}}^{N(t)} \sum_{l:k_l=k'}^{j-1} p_{jl}^{k\ (s)}}{\sum_{k \in K} \sum_{j:r_j \in \{quo,rply\}}^{N(t)} \sum_{l:k_l=k'}^{j-1} p_{jl}^{k\ (s)}} \tag{B.42}$$

Similarly, for parameter ω_k, we have:

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\omega (\delta_{r_l} \gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l} \gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{j=1}^{N(t)} \partial_\omega \left(\delta_{r_j} \gamma_{k_j,k} n_j \cdot (1 - e^{-\omega_k(T-t_j)}) \right) \tag{B.43}$$

Due to the space limit, we will present the equation derivation for each side separately. The left side of the equation is expressed as:

$$\text{Left side} = \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\omega (\delta_{r_l} \gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l} \gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)}} \tag{B.44}$$

$$= \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\delta_{r_l} \gamma_{k_l,k_j} n_l \cdot \left(1 \cdot e^{-\omega_k(t_j-t_l)} + \omega_k \cdot e^{-\omega_k(t_j-t_l)} \cdot (-(t_j - t_l))\right)}{\delta_{r_l} \gamma_{k_l,k_j} n_l \omega_k e^{-\omega_k(t_j-t_l)}} \tag{B.45}$$

$$= \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\cdot (1 - \omega_k (t_j - t_l))}{\omega_k} \tag{B.46}$$

And the right side of the equation is simplified as:

$$\text{Right side} = \sum_{j=1}^{N(t)} \partial_\omega \left(\delta_{r_j} \gamma_{k_j,k} n_j \cdot (1 - e^{-\omega_k (T - t_j)}) \right) \tag{B.47}$$

$$= \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j \left(-e^{-\omega_k (T - t_j)} \cdot (-(T - t_j)) \right) \tag{B.48}$$

$$= \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j (T - t_j) e^{-\omega_k (T - t_j)} \tag{B.49}$$

such that the equation can be further simplified as:

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\cdot (1 - \omega_k (t_j - t_l))}{\omega_k} = \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j (T - t_j) e^{-\omega_k (T - t_j)} \tag{B.50}$$

$$\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k - \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \omega_k (t_j - t_l) = \omega_k \cdot \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j (T - t_j) e^{-\omega_k (T - t_j)} \tag{B.51}$$

$$\omega_k \left[\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k (t_j - t_l) + \sum_{j=1}^{N(t)} \delta_{r_j} \gamma_{k_j,k} n_j (T - t_j) e^{-\omega_k (T - t_j)} \right] = \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \tag{B.52}$$

$$\omega_k \sum_{j=1}^{N(t)} \left[\sum_{l=1}^{N(t_j)} p_{jl}^k (t_j - t_l) + \delta_{r_j} \gamma_{k_j,k} n_j (T - t_j) e^{-\omega_k (T - t_j)} \right] = \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \tag{B.53}$$

$$\omega_k = \frac{\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k}{\sum_{j=1}^{N(t)} \left[\sum_{l=1}^{N(t_j)} p_{jl}^k (t_j - t_l) + \delta_{r_j} \gamma_{k_j,k} n_j (T - t_j) e^{-\omega_k (T - t_j)} \right]} \tag{B.54}$$

$$\approx \frac{\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k}{\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k (t_j - t_l)} \tag{B.55}$$

Therefore, ω_k is updated as follow at iteration $s+1$

$$\omega_k^{(s+1)} = \frac{\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k\,(s)}}{\sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^{k\,(s)} (t_j - t_l)} \tag{B.56}$$

However, δ is a parameter that measures the ability to generate new events (tweets) by a specific tweet type, which should be irrelevant with parameter k, the stance of the tweet. This indicates that δ will be estimated differently given specific stance k. Therefore, the expression of δ should be calculated based on the Q function of the overall intensity $Q(T;\theta)$, not for stance k: $Q_k(T;\theta)$. Taking the partial derivative on parameter θ, we have:

$$\frac{\partial Q(T;\theta)}{\partial \theta} = \frac{\partial \left(\sum_{k\in K} \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \log h_k(t_j - t_l;\theta) \right)}{\partial \theta} - \frac{\partial \left(\sum_{k\in K} \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j;\theta) \mathrm{d}t \right)}{\partial \theta} \tag{B.57}$$

$$= 0 \tag{B.58}$$

such that

$$\frac{\partial \left(\sum_{k\in K} \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \log h_k(t_j - t_l;\theta) \right)}{\partial \theta} = \frac{\partial \left(\sum_{k\in K} \sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k(t - t_j;\theta) \mathrm{d}t \right)}{\partial \theta} \tag{B.59}$$

$$\sum_{k\in K} \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\theta h_k(t_j - t_l;\theta)}{h_k(t_j - t_l;\theta)} = \sum_{k\in K} \sum_{j=1}^{N(t)} \left(\partial_\theta H_k(T - t_j;\theta) - \partial_\theta H_k(0;\theta) \right) \tag{B.60}$$

$$\sum_{k\in K} \sum_{j=1}^{N(t)} \sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\theta (\delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l))}{\delta_{r_l} \gamma_{k_l,k_j} n_l g_k(t_j - t_l)} = \sum_{k\in K} \sum_{j=1}^{N(t)} \partial_\theta \left(\delta_{r_j} \gamma_{k_j,k} n_j (G_k(T - t_j) - G_k(0)) \right) \tag{B.61}$$

For exponential kernel $g_k(t-t_j) = \omega_k e^{-\omega_k(t-t_j)}$, we have:

$$\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l=1}^{N(t_j)} p_{jl}^k \frac{\partial_\theta(\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{k\in K}\sum_{j=1}^{N(t)} \partial_\theta\left(\delta_{r_j}\gamma_{k_j,k}n_j\cdot(1-e^{-\omega_k(T-t_j)})\right) \tag{B.62}$$

For parameter $\delta = \delta_r$:

$$\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^k \frac{\partial_{\delta_r}(\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \partial_{\delta_r}\left(\delta_{r_j}\gamma_{k_j,k}n_j\cdot(1-e^{-\omega_k(T-t_j)})\right) \tag{B.63}$$

Since δ_{r_l} relates to the tweet type of the l-th tweet Z_l, for each specific tweet type r (original tweet, retweet, reply, quote), we have:

$$\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^k \frac{\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)})}{\delta_{r_l}\gamma_{k_l,k_j}n_l\omega_k e^{-\omega_k(t_j-t_l)}} = \sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k}n_j\left(1-e^{-\omega_k(T-t_j)})\right) \tag{B.64}$$

$$\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^k \frac{1}{\delta_{r_l}} = \sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k}n_j\left(1-e^{-\omega_k(T-t_j)})\right) \tag{B.65}$$

$$\frac{1}{\delta_r}\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^k = \sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k}n_j\left(1-e^{-\omega_k(T-t_j)})\right) \tag{B.66}$$

$$\delta_r = \frac{\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^k}{\sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k}n_j\left(1-e^{-\omega_k(T-t_j)})\right)} \tag{B.67}$$

$$\approx \frac{\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^k}{\sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k}n_j} \tag{B.68}$$

Therefore:

$$\delta_r^{(s+1)} = \frac{\sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l:r_l=r}^{N(t_j)} p_{jl}^{k\,(s)}}{\sum_{k\in K}\sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k}{}^{(s)}n_j} \tag{B.69}$$

B.4 Summary

Now we can summarize the EM algorithm derived for the proposed Multivariate Hawkes Point Processes as follow. At E-step, update:

$$p_{jj}^{k\ (s+1)} = \frac{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}}}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)} \tag{B.70}$$

$$= \frac{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}}}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j - t_l)}} \tag{B.71}$$

$$p_{jl}^{k\ (s+1)} = \frac{\delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l g_k(t_j - t_l)} \tag{B.72}$$

$$= \frac{\delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k e^{-\omega_k(t_j - t_l)}}{\mu_k^{(s)} \frac{xe^{-xt_j}}{1-e^{-xT}} + \sum_{l=1}^{j-1} \delta_{r_l}^{(s)} \gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j - t_l)}} \tag{B.73}$$

At M-step, for $k : k \in [1, K]$, update:

$$\mu_k^{(s+1)} = \frac{\sum_{j=1}^{N(t)} p_{jj}^{k\ (s)}}{T} \tag{B.74}$$

$$\gamma_{k',k}{}^{(s+1)} = \frac{\sum_{j:r_j \in \{quo,rply\}}^{N(t)} \sum_{l:k_l=k'}^{j-1} p_{jl}^{k\ (s)}}{\sum_{k \in K} \sum_{j:r_j \in \{quo,rply\}}^{N(t)} \sum_{l:k_l=k'}^{j-1} p_{jl}^{k\ (s)}} \tag{B.75}$$

$$\omega_k^{(s+1)} = \frac{\sum_{j=1}^{N(t)} \sum_{l=1}^{j-1} p_{jl}^{k\ (s)}}{\sum_{j=1}^{N(t)} \sum_{l=1}^{j-1} p_{jl}^{k\ (s)} (t_j - t_l)} \tag{B.76}$$

For parameter δ_r, we have:

$$\delta_r^{(s+1)} = \frac{\sum_{k \in K} \sum_{j=1}^{N(t)} \sum_{l:r_l=r}^{j-1} p_{jl}^{k\ (s)}}{\sum_{k \in K} \sum_{j:r_j=r}^{N(t)} \gamma_{k_j,k} n_j} \tag{B.77}$$

At the end of each iteration $s+1$, update $Q(T;\theta)$:

$$
\begin{aligned}
Q^{(s+1)}(T;\theta) =& \sum_{k\in K}\sum_{j=1}^{N(t)} p_{jj}^{k\ (s)} \log(\mu_k^{(s)} \frac{xe^{-xt}}{1-e^{-xT}}) - \sum_{k\in K}\mu_k^{(s)} \\
&+ \sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l=1}^{j-1} p_{jl}^{k\ (s)} \log h_k^{(s)}(t_j - t_l;\theta) \qquad \text{(B.78)} \\
&- \sum_{k\in K}\sum_{j=1}^{N(t)} \int_{t_j}^{T} h_k^{(s)}(t-t_j;\theta)\mathrm{d}t + C \\
=& \sum_{k\in K}\sum_{j=1}^{N(t)} p_{jj}^{k\ (s)} \log(\mu_k^{(s)} \frac{xe^{-xt}}{1-e^{-xT}}) - \sum_{k\in K}\mu_k^{(s)} \\
&+ \sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l=1}^{j-1} p_{jl}^{k\ (s)} \log\left(\delta_{r_l}^{(s)}\gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j-t_l)}\right) \qquad \text{(B.79)} \\
&- \sum_{k\in K}\sum_{j=1}^{N(t)} \delta_{r_j}^{(s)}\gamma_{k_j,k}^{(s)} n_j \cdot (1-e^{-\omega_k^{(s)}(T-t_j)}) + C \\
\approx& \sum_{k\in K}\sum_{j=1}^{N(t)} p_{jj}^{k\ (s)} \log(\mu_k^{(s)} \frac{xe^{-xt}}{1-e^{-xT}}) - \sum_{k\in K}\mu_k^{(s)} \\
&+ \sum_{k\in K}\sum_{j=1}^{N(t)}\sum_{l=1}^{j-1} p_{jl}^{k\ (s)} \log\left(\delta_{r_l}^{(s)}\gamma_{k_l,k_j}^{(s)} n_l \omega_k^{(s)} e^{-\omega_k^{(s)}(t_j-t_l)}\right) \qquad \text{(B.80)} \\
&- \sum_{k\in K}\sum_{j=1}^{N(t)} \delta_{r_j}^{(s)}\gamma_{k_j,k}^{(s)} n_j + C
\end{aligned}
$$

Each time we compare $Q^{(s)}$ and $Q^{(s+1)}$ and stop the loop when:

$$
|Q^{(s+1)}(T;\theta) - Q^{(s)}(T;\theta)| \le \epsilon \qquad \text{(B.81)}
$$