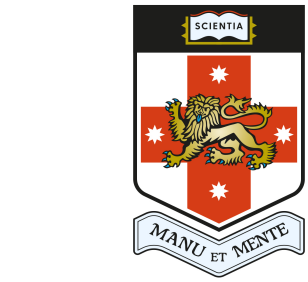

UNSW
SYDNEY

Predicting the popularity of tweets using the theory of point processes

Wai Hong Tan
School of Mathematics and Statistics
University of New South Wales

A thesis in the fulfilment of the requirements for the degree of
Doctor of Philosophy
March 2019

Thesis/Dissertation Sheet

Surname or Family name	:	Tan
First name	:	Wai Hong
Abbreviation for degree as given in the University calendar	:	PhD
Faculty	:	Science
School	:	Mathematics and Statistics
Title	:	Predicting the popularity of tweets using the theory of point processes

Abstract 350 words maximum

This thesis focuses on the problem of predicting the tweet popularity, or the number of retweets stemming from an original tweet. We propose several prediction methodologies using the theory of point processes, where the prediction of the future popularity of a tweet is based on observing the retweet time sequence up to a certain censoring time, and the prediction performance is evaluated on a large Twitter data set.

We first propose a marked point process model, termed the Marked Self-Exciting Process with Time-Dependent Excitation Function, or the MaSEPTiDE for short. The intensity process of the model is interpretable as a cluster Poisson process, which implies that the model can be simulated using the cascading algorithm similar to that used for the efficient simulation of Hawkes processes, and the prediction can be done properly by exploiting the probabilistic properties of the model. The MaSEPTiDE approach shows highly accurate tweet popularity predictions compared to state-of-the-art approaches, especially at shorter censoring times.

We further propose an inhomogeneous Poisson process model and an estimation method which utilizes internal and external knowledge, based on the times of historical retweets up to the censoring time, and the complete retweet sequences in the training data set respectively. The knowledge is combined using a novel empirical Bayes type approach, where the prior distribution for the model parameter is constructed based on the external knowledge, and the likelihood is calculated based on the internal knowledge. The mode of the posterior distribution is used as the estimator of the finite-dimensional parameter, and suitable functionals of the predictive distribution for the number of retweets implied by the estimated model are used to predict the tweet popularity. The model, termed the EB Poisson model, is found to be both efficient and accurate, with an additional advantage of being able to predict without observing any retweets.

The proposed EB approach of inference is applicable on other point process models, such as the MaSEPTiDE model, to improve the prediction performance and computational efficiency. We demonstrate this by applying the EB approach on the MaSEPTiDE model and reporting further improvements in the prediction accuracy.

FOR OFFICE USE ONLY Date of completion of requirements for Award:

ORIGINALITY STATEMENT

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Signed ……..

Date ……..

INCLUSION OF PUBLICATIONS STATEMENT

UNSW is supportive of candidates publishing their research results during their candidature as detailed in the UNSW Thesis Examination Procedure.

Publications can be used in their thesis in lieu of a Chapter if:

- The student contributed greater than 50% of the content in the publication and is the "primary author", i.e. the student was responsible primarily for the planning, execution and preparation of the work for publication
- The student has approval to include the publication in their thesis in lieu of a Chapter from their supervisor and Postgraduate Coordinator.
- The publication is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in the thesis

Please indicate whether this thesis contains published material or not.

☐ *This thesis contains no publications, either published or submitted for publication (if this box is checked, you may delete all the material on page 2)*

☒ *Some of the work described in this thesis has been published and it has been documented in the relevant Chapters with acknowledgement (if this box is checked, you may delete all the material on page 2)*

☐ *This thesis has publications (either published or submitted for publication) incorporated into it in lieu of a chapter and the details are presented below*

CANDIDATE'S DECLARATION

I declare that:

- I have complied with the Thesis Examination Procedure
- where I have used a publication in lieu of a Chapter, the listed publication(s) below meet(s) the requirements to be included in the thesis.

Name	**Signature**	**Date (dd/mm/yy)**

Abstract

This thesis focuses on the problem of predicting the tweet popularity, or the number of retweets stemming from an original tweet. We propose several prediction methodologies using the theory of point processes, where the prediction of the future popularity of a tweet is based on observing the retweet time sequence up to a certain censoring time, and the prediction performance is evaluated on a large Twitter data set.

We first propose a marked point process model, termed the Marked Self-Exciting Process with Time-Dependent Excitation Function, or the MaSEPTiDE for short. The intensity process of the model is interpretable as a cluster Poisson process, which implies that the model can be simulated using the cascading algorithm similar to that used for the efficient simulation of Hawkes processes, and the prediction can be done properly by exploiting the probabilistic properties of the model. The MaSEPTiDE approach shows highly accurate tweet popularity predictions compared to state-of-the-art approaches, especially at shorter censoring times.

We further propose an inhomogeneous Poisson process model and an estimation method which utilizes internal and external knowledge, based on the times of historical retweets up to the censoring time, and the complete retweet sequences in the training data set respectively. The knowledge is combined using a novel empirical Bayes type approach, where the prior distribution for the model parameter is constructed based on the external knowledge, and the likelihood is calculated based on the internal knowledge. The mode of the posterior distribution is used as the estimator of the finite-dimensional parameter, and suitable functionals of the predictive distribution for the number of retweets implied by the estimated model are used to predict the tweet popularity. The model, termed the EB Poisson model, is found to be both efficient and accurate, with an additional advantage of being able to predict without observing any retweets.

The proposed EB approach of inference is applicable on other point process models, such as the MaSEPTiDE model, to improve the prediction performance and computational efficiency. We demonstrate this by applying the EB approach on the MaSEPTiDE model and reporting further improvements in the prediction accuracy.

Acknowledgements

I am truly grateful to:

- my main supervisor Dr. Feng Chen, for his patient guidance, encouragement and generosity with his time and expertise throughout my PhD studies.
- my parents, for the love and compassion, care and support bestowed upon me for my entire life.
- my peers Zhi Yee Chng, Yeap Shong Mei, Vincent Chin, Kevin Limanta, Sabarina Shafie, and Flora Zengyan Fan, for always helping me and keeping my spirits high.
- my sponsor, the Ministry of Higher Education, Malaysia, for funding me throughout my research period.
- my university and the Australian government, for granting me access to the high performance computational clusters.

Contents

Chapter 1

Introduction

The rapid technological advancements in the recent years have offered worldwide connectivity to the internet, making information sharing much more accessible and economical. Such developments have facilitated the progressive transition of the internet to its contemporary phase of Web 2.0 (DiNucci, 1999; O'Reilly, 2005) which emphasizes the interactivity between creators of web contents, in contrast to its predecessor Web 1.0 where people were restricted to passive viewing of the contents. The shifted attention of the online community to user-generated contents has therefore promoted the growth of social media which encompasses collaborative projects, blogs, social networking sites, virtual game worlds, and virtual social worlds (Kaplan and Haenlein, 2010).

As of year 2018, social media has garnered over 2.5 billion users[1] across the globe, accounting for roughly a third of the entire Earth's population, and overwhelming the influence from that of the mainstream media. Moreover, approximately three quarters of the internet users are actively engaged in social networking sites, fostering them to be the fastest growing form of social media. Social networking sites are revolutionary channels of information dissemination democratized by vastly diverse users who indefinitely generate and distribute their own contents, thereby triggering an explosive growth of information. These user-generated contents are presented in the form of textual, visual, or aural information, and serve as the primary constituents of the internet traffic.

The phenomenal influence exerted by user-generated contents as such has motivated researchers to explore them from various perspectives. This chapter aims to give some introductory remarks specific to a social networking site known as Twitter. We highlight the background of Twitter by discussing some of its statistical and topological features in Section 1.1. Then, we present several significant studies and useful applications based on Twitter in Section 1.2. Finally, we illustrate the

[1] from https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/

structure of a typical tweet, and describe the Twitter data set used throughout this thesis in Section 1.3.

1.1 Twitter Background

Microblogs are social networking sites which integrate the features of instant messaging and blogging, with certain restrictions imposed such as the character limit of messages. Twitter[2] is a quintessential microblogging platform where users share information in the forms of 140-character[3] messages called tweets. By the year of 2018, the platform has accumulated over 300 million users, boasting an average of 6,000 tweets per second, which equates to over 350,000 tweets per minute or 500,000 million tweets per day[4], making it the most popular microblogging website to date.

The Twitter network consists mainly of followee-follower relationships which result from the acts of following or being followed by other users. Such relationships require no reciprocation, implying that a user can follow any other users, and the user being followed needs not follow back. An early exploratory work of Java et al. (2007) concluded that the Twitter network exhibits high degree of correlation and reciprocity, indicating close mutual acquaintances among the users. They highlighted that it is important to understand the intention of Twitter users by analyzing the aggregate behaviours across the communities so that useful features can be incorporated into the platform interface to potentially attract more users. As Twitter becomes increasingly more popular over the subsequent years, the study conducted by Kwak et al. (2010) reached a rather different conclusion. They noticed that the followee-follower relationships on Twitter have surprisingly low reciprocity, and reciprocated users tend to be homophilous.

More recently, Newman (2017) reported that the attention dynamics online are frequently dominated by a diverse array of decentralized non-elite users, based on a climate change assessment report publicized on Twitter. Their finding was contradictory to the research conducted by the Yahoo! company in year 2011, where about half of all tweets on the Twitter network were purportedly generated by 20,000 elite users comprising of media outlets and celebrities (Wu et al., 2011). This warrants that ordinary users are also substantially influential to transfer information on the network, and should be accounted for when modelling the information diffusion on Twitter. Both the findings of Kwak et al. (2010) and Newman (2017) have evinced that the Twitter network tends to evolve over time, primarily due to the increased heterogeneity of Twitter users.

Furthermore, persistent and headline news are known to diffuse through the

[2]https://twitter.com

[3]valid as of the third quarter of 2017, but the character limit gets doubled after that

[4]from www.internetlivestats.com/twitter-statistics/

Twitter network massively (Kwak et al., 2010), making it a platform of ambient journalism (Hermida, 2010). The real-time nature of the Twitter platform, with the limitation on its content length, makes conveying information to a vast panoply of audience both concise and effective. Because of its exceptional quality to proliferate information, the platform has been used during major incidents to quickly spread messages to targeted users. As an indicative example, Hughes and Palen (2009) examined how Twitter was used for political conventions and notifications of natural disasters, and concluded that tweets sent during such times tend to reveal features of information dissemination that support information broadcasting and brokerage.

Admittedly, Twitter has provided its users social gratification by satisfying their various needs, in particular the feeling of security resulting from the interconnectedness with other users. It has enabled its users to share their individual thoughts while engaging in communal activities at the same time, making it simultaneously individualistic and communal (Murthy, 2018). From a broader perspective, besides propagating information from within itself, Twitter also serves as an intermediary platform to transmit information externally so as to reach the manifold of internet users in other online communities.

1.2 Twitter Studies and Applications

Twitter, with its application programming interface to crawl the network and its mechanism to relay information, has offered unprecedented opportunities for computer scientists, sociologists, linguists, and physicists to conduct research based on the platform. It has inspired a multitude of compelling studies, such as evaluating the likelihood of retweets based on the interestingness of content (Naveed et al., 2011), predicting if a tweet will be retweeted based on social and tweet features (Petrovic et al., 2011), estimating the rise and fall of influence propagation (Matsubara et al., 2012), forecasting the trend of future retweets (Gupta et al., 2012; Zhang et al., 2013), and modelling the random series of events based on tweet hashtags (Alves et al., 2016). The dynamics of retweets stemming from a root tweet have also been modelled, for instance by Kumar et al. (2010) and Nishi et al. (2016), which then motivate the work of Aragón et al. (2017) reviewing various statistical models for threaded online discussions originating from different social media platforms. Some studies are more application-based, and depend mainly on textual analysis or tweet popularity prediction, highlighted as follows.

1.2.1 Sentiment Analysis

Twitter studies frequently revolve around analyzing the sentiments of its contents, which refers to the process of computationally identifying and categorizing textual

opinions, usually with the aim to understand the writer's attitudes or perceptions towards some items. This is especially useful for consumers who want to assess the sentiment of a specific product prior to making a purchase, or companies who want to monitor the public sentiments of their brands.

The most commonly used feature in *sentiment analysis* is the n-grams. Briefly, an n-gram consists of a sequence of items from a collected sample of text or speech corpus. The intuition behind n-gram is to capture the linguistic structure from the statistical point of view, and predict the letter or word following a given one. Going by the conventional number prefixes, unigram, bigram, trigram and so on respectively denote $n = 1, 2, 3, \ldots$ subsequent characters, the optimal length by which depends largely on its application; see Hasan et al. (2007) for an in-depth discussion of the n-gram.

By using several machine learning algorithms such as the maximum entropy classifier, support vector machine, and naive Bayes, Go et al. (2009) achieved high accuracy in classifying the sentiments of tweets. Specifically, they performed distant supervised learning from the aforementioned n-gram features and classified messages as being positive or negative with respect to a query term. Following this, Pak and Paroubek (2010) added a neutral class of sentiment to the algorithm, based on the multinomial version of similar naive Bayes classifier, and features resembling that of Go et al. (2009). More recently, Vosoughi et al. (2016) proposed an enhancement over purely linguistic classifiers, through employing a Bayesian approach which combines the n-gram linguistic features with spatial, temporal, and author-related contextual information.

Sentiment analysis on Twitter is useful in numerous real-life applications. Tumasjan et al. (2010) considered Twitter a platform used for political deliberation and analyzed tweet sentiments by machine learning to forecast the results of elections. Sakaki et al. (2010) devised a classifier based on the semantic features of tweets to detect earthquakes in Japan, with a probabilistic spatio-temporal model to find the epicenters of the earthquakes, and an efficient system to notify potential victims upon the detection of an event. Analyzing tweets for keywords or sentiments has various applications in finance as well, for example to predict stock market indicators by measuring the collective mood on the platform. Zhang et al. (2011) postulated that the emotional tantrum on Twitter is correlated to how the stock market will be doing the next day. By using mood words such as hope, fear, and worry as the emotional tags of tweets, they counted the number of tweets containing such words and used them to predict the behaviours of stocks the next day. With the same objective in mind, Bollen et al. (2011) measured the mood with extra dimensions, and by identifying dimensions that are Granger causative to the prices, such as calm and happy, they reported an improvement in the prediction accuracy of stock prices

compared to that of Zhang et al. (2011).

1.2.2 Popularity Prediction

For introductory purposes we shall briefly discuss some of the important works on tweet popularity prediction here, and present more details in Chapter 3. We have mentioned in Section 1.1 that the message posted on Twitter is referred to as a *tweet*. As a tweet is posted, it may be shared by the followers of the tweeting account through an action known as retweeting, which explicitly refers to the tweet via its unique identification number, and results in a *retweet*. This retweeting process can iterate indefinitely, forming a cascade of retweets. When studying information diffusion on Twitter, it is often of interest to predict the tweet *popularity*, which is naturally measured by the total number of retweets stemming from the original tweet.

One noticeable work of tweet popularity prediction is the Bayesian approach model of Zaman et al. (2014), which requires the complete network information to be operable. Models based on the theory of point processes (Zhao et al., 2015; Kobayashi and Lambiotte, 2016), which do not require such information, were also shown to have good prediction performances. The model proposed by Mishra et al. (2016), on the other hand, combines point process models with feature-based approaches to predict the tweet popularity. Other models like the growth-adoption model of Lymperopoulos (2016), the spatio-temporal heterogeneous Bass model of Yan et al. (2016), and the concept drift model of Li et al. (2016), were all proposed for tweet popularity predictions.

Prediction models employed on other social media platforms are also relevant as the proposed methodologies may be applicable to tweet popularity prediction. Notably, Agarwal et al. (2009) proposed a dynamic linear regression model to predict the click-through rate for Today Module on Yahoo! Front Page. Activities on other platforms like Youtube were also modelled, for instance using linear regression models (Szabo and Huberman, 2010; Pinto et al., 2013) and classification models (Gürsun et al., 2011; Ahmed et al., 2013; Figueiredo, 2013). Other closely related works include the reinforced Poisson model applied on Sina Weibo (Gao et al., 2015) and the model by Wu et al. (2016) that incorporates temporality and seasonality, applied on Flickr image data set.

Popularity predictions of online contents in general have been used extensively in web distribution systems. Specifically, popular web items can be prefetched into mobile users' cache from colocated peers to offload mobile data (Han et al., 2012). This implies that if the popularity of a web item has been efficaciously predicted, future content requests can simply rely on the colocated mobile users, thereby re-

ducing the network traffic and battery consumption of mobile phones. The load of data traffic can also be reduced by predicting users who will potentially trigger the request of a web content (Galuba et al., 2010), or by proactively pushing the content to users prior to the request (Malandrino et al., 2012). The eviction of items from the cache can also be optimized by accurately predicting the future demand of a specific content, which in turn minimizes the wastage of bandwidth (Famaey et al., 2011).

Twitter-oriented popularity prediction, on the other hand, has numerous other useful applications (Yu and Kak, 2012). Besides helping the platform itself to rank contents more effectively, it is also useful in estimating movie revenues (Asur and Huberman, 2010), approximating the citation counts of research articles (Eysenbach, 2011), assisting marketing firms to maximize their revenues through optimal placements of advertisements (Yang and Leskovec, 2011), and serving as a proxy to political candidates in election campaigns (Van Aelst et al., 2017).

As a remark, we note that tweet popularity has also been measured differently by the number of users opportune to see the tweet in their news feeds, synonymously referred to as the number of shows or the audience size (Kupavskii et al., 2013). While it is desirable for most contents tweeted to receive as many retweets as possible, a new brand launching an advertising campaign might find it more sensible to receive as many shows of tweets with its name as possible to increase the brand awareness. Predicting the popularity based on this context can then assist the advertising firm to estimate the initial costs involved to achieve the desired popularity level.

Besides the applications based on tweet sentiment analysis and popularity prediction, there are other useful applications sourcing from Twitter. The work from Hughes and Palen (2009) for instance, mentioned how emergency management could use Twitter and similar microblogging platforms to deliver warnings during unforeseen circumstances like response and recovery situations. On the other hand, Bakshy et al. (2011) quantified the influences of Twitter users based on their attributes to assist marketers and planners in spreading information more effectively through the identified influencers on the network. Finally, the prevalence of geolocation feature makes it possible to study the global mobility patterns of users who have registered for the service (Hawelka et al., 2014), which provides invaluable insights on tourist activities, migration flows, and contagion of diseases.

1.3 The Tweet Structure and Data

Before we proceed to presenting the Twitter data set used throughout the rest of this thesis, it would be beneficial to first preview the structure of a tweet. Figure 1.3.1

shows the structure of a typical tweet[5], followed by the description for each of its component. For the ease of interpretation, the tweet poster shall be referred to as a tweeter, individual accounts retweeting the tweet as retweeters, and the remaining accounts as viewers.

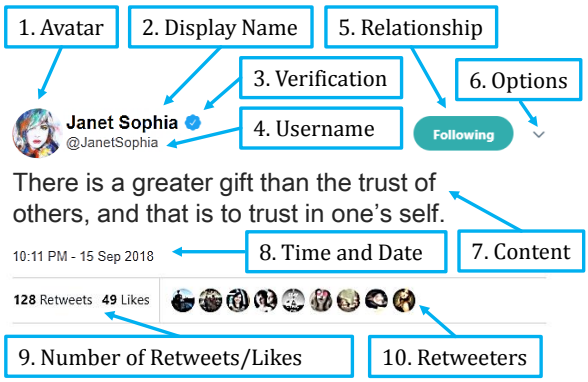

Figure 1.3.1: The structure of a tweet. Each component of the tweet has been labelled for convenience. The tweet was posted by a user called Janet Sophia on 15^{th} September 2018, 10:11 PM. The tweet was retweeted 128 times and liked 49 times.

1. Avatar: the tweeter's profile photo that is usually representative of the tweeter.
2. Display name: typically contains the full/real name of the tweeter.
3. Verification: proves that Twitter has verified the account of public interest.
4. Username: the unique Twitter handle, used to identify/mention the tweeter.
5. Relationship: shows if the tweeter is being followed by the tweet viewer.
6. Options: contains several options for the viewer including to embed, report, or copy the link of the tweet.
7. Content: the main body of the tweet with a maximum of 140/280[6] characters, which may contain mentions (@), hashtags (#), or URLs (often shortened) aside from generic texts.
8. Time and date: the posting time and date of the tweet, adjusted according to the viewer's time zone.
9. Number of retweets/likes: the number of retweets/likes that the tweet has attracted up to the viewing time.
10. Retweeters: a list showing viewers who have retweeted the tweet.

[5]the tweet has been synthetically generated for demonstrative purposes
[6]the character limit has been doubled since late 2017

By clicking on either the avatar or the display name, the tweeter's profile page will show up, revealing more information such as the number of users following, or is followed by, the tweeter. The unique Twitter handle, or the username, on the other hand, is useful for numerous Twitter analytics[7] to provide fine-grained details such as the join time and date or the activity levels of any given user. While the tweet content is useful for works manipulating its semantic features as discussed in Section 1.2.1, sophisticated machine learning algorithms or classifiers are frequently required. In contrast, the time and date, the number of retweets or likes, and the list of retweeters require minimal preprocessing and are generally more informative to predict the future activities of tweets.

With the tweet structure clarified, we present herein the Twitter data set which had motivated our modelling and prediction methodologies in Chapter 4-6. The data[8] was intially collected by Zhao et al. (2015) and contains a total of 166,069 reasonably popular tweets published in 2011 from October 6 (06:00 UTC) to November 6 (06:00 UTC), each with at least 49 retweets within seven days of publishing. For each tweet, the data includes its unique tweet identification number, its tweet and retweet times within seven days of its publication, and the numbers of followers of its tweeter and its retweeters. Note that the data lacks the complete Twitter network information, that is, for a retweet, the data only has its publishing time and the number of followers of the retweeting account, without information on whether the original tweet or any previous retweet is being retweeted. Thus, methodologies which assume the complete Twitter network information, such as that of Zaman et al. (2014), does not apply here. Following Zhao et al. (2015), the 71,815 tweets published in the first seven days of the study period shall be referred to as the training data, and the remaining 94,254 tweets published in the next eight days shall be referred to as the test data. See Figure 1.3.2 for five randomly selected retweet cascades/retweet time sequences from the training data set, and Table 1.3.1 for the corresponding information.

We have plotted the numbers of followers in Figure 1.3.2 on the log scale as these numbers are considerably large for the majority of the tweeters or retweeters in the data set. It can be observed from Figure 1.3.2 that the retweets tend to occur in clusters or bursts. This suggests that self-exciting processes are potentially suitable for such data. For that, we have formulated a marked self-exciting point process model to capture the retweeting dynamics and predict the future popularity of tweets, discussed in Chapter 4.

Next, Table 1.3.1 shows the publishing time and date for each tweet in Figure 1.3.2, together with the tweet and retweet times in unit seconds, and the num-

[7] for example https://foller.me/

[8] from http://snap.stanford.edu/seismic/

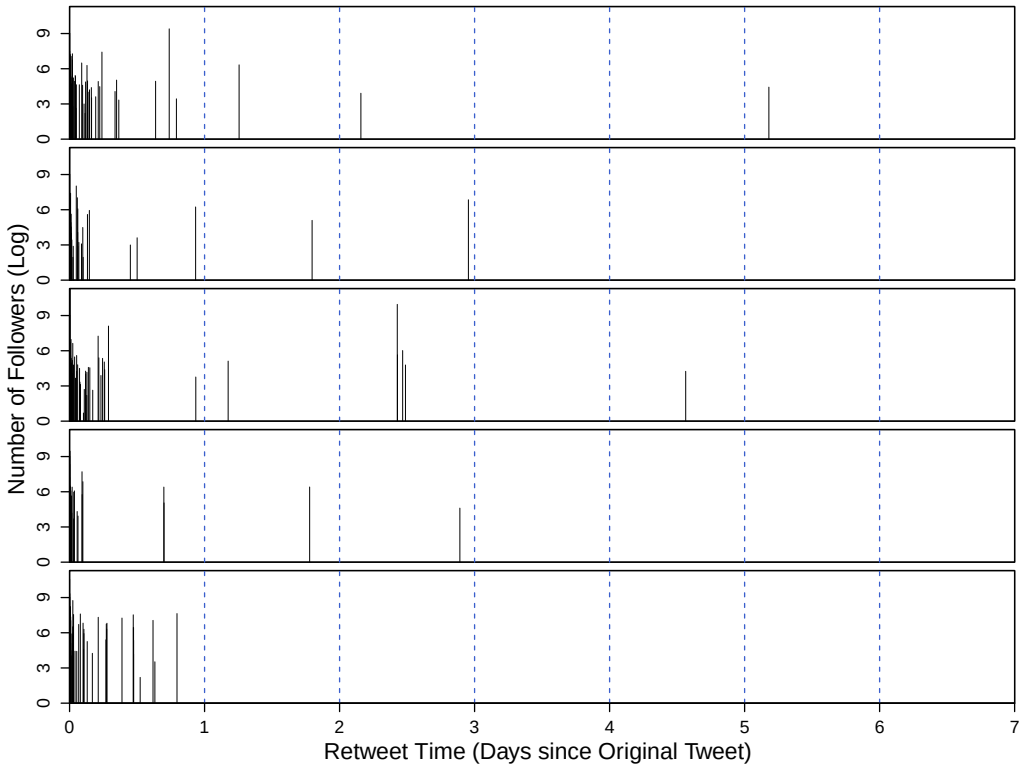

Figure 1.3.2: Times of retweets and the corresponding numbers of followers of the retweeting accounts on the log scale, for five randomly selected retweet cascades from the training data set. For all the retweet cascades, the retweets tend to occur in clusters, especially near the publication times of the original tweets.

bers of followers of the corresponding tweeter and retweeters. Note that all the cascades start with the original tweets at time zero relative to the actual posting times of the day, where subsequent retweet times are orderly arranged in an increasing manner. The numbers of followers are naturally attached to these times, and so the first observation for each retweet cascade accounts for the number of followers of the original poster/tweeter and the remaining numbers are of the retweeters.

The first retweet in sample cascade 1 of Table 1.3.1 occurred almost instantaneously, that is, 11 seconds after the original tweet was published. This can be attributed to the large number of followers of the original poster, counting at a staggering 283,215. The effect attributable to the number of followers is also exhibited by the rest of the sample cascades, where a larger number generally results in shorter waiting time until the arrival of the first retweet. Also, most of the retweet cascades available in the data originate from tweeters with considerably large numbers of followers, which is consistent with the finding of Wu et al. (2011) regarding the domination of elite users on the network.

Based on the time and date of a tweet, we can further calculate the relative posting time in unit days, which accounts for the time passed since the start date of collection, with the integer part denoting the number of days removed. For

Table 1.3.1: The posting time and date, the tweet and retweet times, and the numbers of followers of the tweeter and retweeters, for each retweet cascade in Figure 1.3.2. For sample cascade 1, the first retweet occurred 11 seconds after the original tweet was posted, the swift response by which can be attributed to the large number of followers of the original poster/tweeter.

Sample cascade	Time (UTC)	Date (dd-mm)	Tweet and retweet times	Numbers of followers
1	03:07:33	09-10	0,11,. . . ,447527	283215,369,. . . ,84
2	02:54:38	08-10	0,70,. . . ,255232	122,330728,. . . ,943
3	22:09:55	09-10	0,37,. . . ,394301	12037,80,. . . ,70
4	01:50:00	10-10	0,17,. . . ,249778	114755,221,. . . ,100
5	19:12:34	08-10	0,55,. . . ,68745	8835,38592,. . . ,2074

example, sample cascade 1 in Table 1.3.1 was tweeted 2 days, 21 hours 7 minutes and 33 seconds after the start date on 6^{th} October 6:00 UTC, which equates to 2.880 days or the relative posting time of 0.880 days. The relative posting times for the remaining sample cascades can be calculated similarly, yielding the values of $0.871, 0.674, 0.826$, and 0.550 days respectively. Such measurement of posting time is useful for models built based on the diurnal patterns of humans' activity levels, as we shall see in Chapter 5.

On another note, the empirical cumulative distribution of the retweet times is shown in Table 1.3.2. A close scrutiny reveals that approximately half of the total numbers of retweets have accumulated within one hour since the publications of the original tweets, thereby exhibiting the transient nature of tweets (Bray, 2012; Rey, 2014). The severely right-skewed distribution of the retweet times also seems compatible with the heavy-tailed distributions of human response times in other activities like e-mail correspondence (Malmgren et al., 2008). Such observation has motivated us to use a heavy-tailed function, for example the power-law function, when modelling the variation of the retweet intensity over time.

Table 1.3.2: The percentages of retweets that occurred up to each censoring time in the training data. The majority of retweets have happened in the first 12 hours.

Censoring time (hours)	1	2	3	4	5	6	12	168
% of retweets	51.1	59.1	63.8	67.1	69.6	71.6	79.0	100.0

It might also be of interest to have some insights on the sizes of the retweet cascades. For that, we show in Table 1.3.3 the summary statistics for the actual popularity values towards the end of the observation period of seven days, or the *final popularity*, for both the training and test data sets. By the first, second, and third quartiles (Q_1, Q_2, and Q_3), both data sets seem to have nearly identical distributions of final popularity. The similarities between both data sets imply some degree of homogeneity, which is important in the modelling process so that a model can be

Table 1.3.3: The summary statistics for the actual final popularity values, with Q_1, Q_2 and Q_3 denoting the first, second, and third quartiles respectively. The values are nearly identical for both the training and test data sets.

	Min	Q_1	Q_2	Q_3	Max	Mean
Training	49	70	109	216	33484	205.5
Test	49	70	110	222	17183	210.7

built based on the training data set and the performance can, in turn, be evaluated based on the test data set.

To facilitate our discussion in later chapters, for each retweet cascade we denote the original tweet time by $\tau^0 = 0$, and the subsequent retweet times by $\tau_i, i = 1, 2, \ldots$, relative to the posting time of the original tweet t^0. That is, $\tau^0 < \tau_1 < \tau_2 < \ldots$. Furthermore, we denote the number of followers of the original tweeter by n^0 and the numbers of followers of subsequent retweeters by $n_i, i = 1, 2, \ldots$. The superscripts have been used for τ^0, n^0, and t^0 to account for the readily available information as soon as a tweet is posted.

Chapter 2

Point Processes

Point processes are stochastic processes whose realizations consist of point events in time or space which have been extensively studied due to their wide applicability in various fields. Probabilistic models formulated based on such processes have been abundantly proposed in the recent years, many of which aiming to accurately predict the popularity of tweets. In relation to our discussion and model formulations in later chapters, we shall present herein a brief introduction to some of these processes.

The inhomogeneous Poisson process and self-exciting point process are frequently used to capture the retweeting dynamics and predict the future popularity of tweets. The former is preferred for its convenient mathematical properties, and the latter for its attribute where the past instances of observed events tend to make future occurrences of events more probable. Self-excitation is an especially useful feature in modelling the retweet activities on Twitter as posts rapidly going viral tend to get retweeted more by interconnected users on the network. Based on the forms assumed by these processes, the estimations of model parameters and predictions of the future retweet volumes can be properly implemented.

This chapter aims to provide introductory remarks to the theory of point processes by explaining the fundamental concepts used in this thesis. We commence by giving flavour to the definitions of point processes in Section 2.1. We discuss the properties of the Poisson processes in Section 2.2, introduce the concept of conditional intensity in Section 2.3, and present details on the self-exciting point process in Section 2.4. We show how parameter estimation can be done for any given time sequence from the derived likelihood function in Section 2.5, and discuss how the goodness-of-fit of a point process model can be assessed in Section 2.6. We demonstrate how the Poisson processes can be simulated efficiently in Section 2.7, and finally how the predictions of future events based on different processes can be made in Section 2.8.

2.1 Definitions and Interpretations

We shall focus primarily on *temporal point processes* in this thesis, since the retweet events are distributed over the positive half-line $\mathbb{R}_+$ along the time axis. The temporal point process can be defined as a random sequence of points $\tau_1, \tau_2, \ldots \in \mathbb{R}_+$, with the associated counting process,

$$N(t) = \sum_{i=1}^{\infty} \mathbb{1}\left\{\tau_i \leqslant t\right\}, t \geqslant 0,$$

where $N(t) := N((0, t])$ counts the number of events from time zero up to time t, and is piecewise constant with a jump size of one at times τ_i. It is also convenient sometimes to interpret the point process N as a *random measure* via $N(B) = \sum_{i=1}^{\infty} \delta_{\tau_i}(B)$ for all measurable set B, where δ_{τ_i} is the *Dirac measure* defined by

$$\delta_{\tau_i}(B) = \mathbb{1}\left\{\tau_i \in B\right\} = \begin{cases} 1 & \text{if } \tau_i \in B \\ 0 & \text{otherwise.} \end{cases}$$

The sequence of event times can be accompanied by certain random variables with some degree of influence on the process, called the event marks. Such marks can take some diverse forms, including integers, real numbers, lines, geometrical objects or even other point processes (Moller and Waagepetersen, 2003), and are often assumed to be independent of each other and identically distributed (i.i.d). Our discussion in this chapter generalizes to point processes with marks, or *marked point processes*, but we shall be conservative at this point and describe the processes without involving marks, as they will be elucidated in later chapters.

2.2 The Poisson Process

The Poisson process is a subclass of point process which supports the more complex formulations of point processes such as the Hawkes process. Specifically, if the sequence of interevent times $\tau_i - \tau_{i-1}$ for $i = 1, 2, \ldots$ are i.i.d exponential random variables with mean $1/\lambda$, then the process is a *homogeneous Poisson process* with rate λ. In this case, if we denote by $N(a, b)$ the number of points in the half-open interval $(a, b]$ for $0 \leqslant a < b$, the probability of having x points in the interval with mean $\lambda(b - a) = \int_a^b \lambda \, \mathrm{d}s$ is,

$$\Pr\left\{N(a, b) = x\right\} = \frac{[\lambda(b - a)]^x}{x!} e^{-\lambda(b-a)}.$$

In contrast, when the rate of event arrival varies with time, the *inhomogeneous Poisson process* with a time-dependent function $\lambda(t)$ will be useful. In this case,

the probability of having x points in the interval $(a, b]$ for $0 \leqslant a < b$ with mean $\Lambda(a, b) = \int_a^b \lambda(s)\,\mathrm{d}s$ is,

$$\Pr\{N(a,b) = x\} = \frac{[\Lambda(a,b)]^x}{x!} e^{-\Lambda(a,b)}.$$

The Poisson process has the property that each point is stochastically independent to all the other points in the process, and is occasionally referred to as a purely or completely random process (Daley and Vere-Jones, 2003). Nonetheless, the inherent nature of the process implies that it does not adequately describe phenomena in which there are sufficiently strong interactions between the points. This implies that other point processes, such as the self-exciting point process, might be suitable to capture the interactions. Before we proceed to presenting the self-exciting point process, it helps to first present the concept of conditional intensity.

2.3 Conditional Intensity

A temporal point process can be considered a model for an evolving stochastic system which may depend on the historical events in a certain way. The idea requires a proper definition of the history that has to reflect, at any time t, the accumulated information up to that time point. Therefore, we define

$$\begin{aligned} \mathcal{F}_t &:= \sigma\{N(s), 0 < s \leqslant t\} \\ \mathcal{F}_{t-} &:= \sigma\{N(s), 0 < s < t\}, \end{aligned}$$

where the system $\{\mathcal{F}_t, t \geqslant 0\}$ represents the dynamic evolution of a point process N, and is the *natural filtration* generated by N. The notation $\mathcal{F}_t$ can be interpreted as the process history at time t, and $\mathcal{F}_{t-}$ as the prior-t history.

A point process N can be conveniently characterized by its *(conditional) intensity function* $\lambda(t),\ t \geqslant 0$, where $\lambda(t)$ can be defined as the instantaneous event rate at time t given the prior-t history, that is,

$$\lambda(t) = \lim_{h \to 0} \frac{\Pr\{N([t, t+h)) = 1 | \mathcal{F}_{t-}\}}{h}. \tag{2.3.1}$$

The conditional intensity in (2.3.1) is a rather naive definition, but a mathematically more precise formulation requires the theory of martingales, detailed for example in Section 8.3 of Last and Brandt (1995). For an inhomogeneous Poisson process with rate function $\lambda(t)$, it is easy to see that its intensity process is deterministic and equals $\lambda(t)$, $t \geqslant 0$, which does not depend on its history. However, the intensity of certain point processes, such as the self-exciting point process detailed next in Section 2.4, can be dependent on its history.

2.4 Self-Exciting Point Process

The *self-exciting point process* is a process where the event arrival rate depends on instances from the past. Such effect is typically governed by a memory kernel function where the cumulative effects of all previous instances are accounted for, with the most recent event exerting the greatest influence. When triggered by such excitation effect, the arrival of each event will inflate the conditional intensity, thereby making future arrivals of events more probable.

Figure 2.4.1 depicts ten sample events[1] in terms of the event times, counting process, and the conditional intensity. The upper panel shows that the events tend

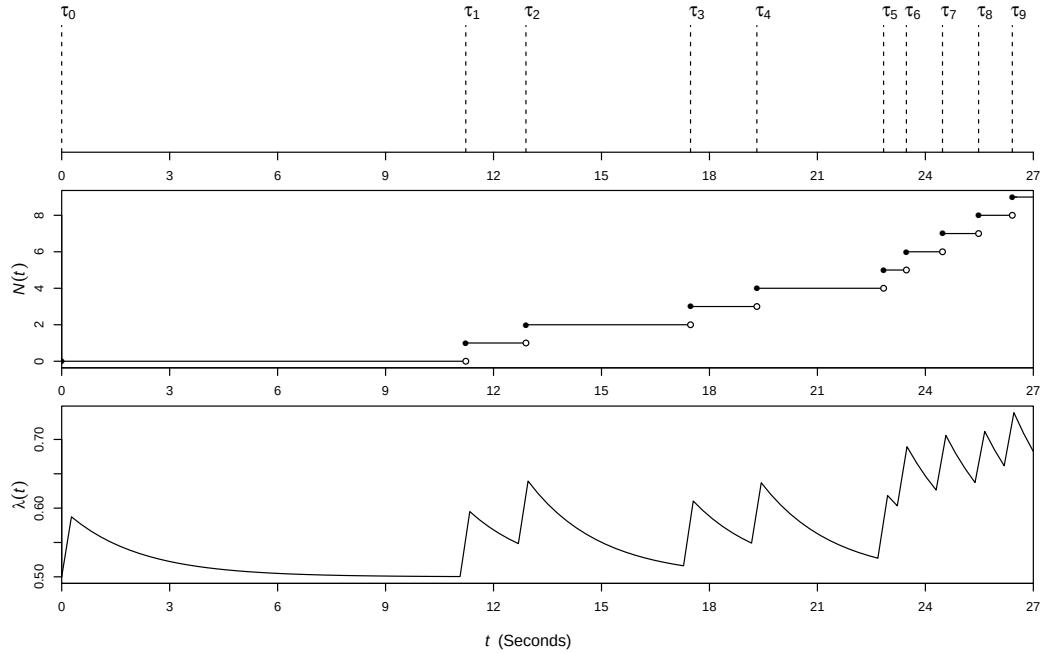

Figure 2.4.1: Ten sample events with different representations, starting at $\tau^0 = 0$ and followed by $\tau_1 < \tau_2 < \ldots < \tau_9$. The upper panel shows the times of such activities, the middle panel shows the counting process $N(t)$ with the increment of one every time an event is observed at time t, and the bottom panel shows the conditional intensity $\lambda(t)$ based on a memory kernel function that decays exponentially with time t.

to occur in clusters or bursts, suggesting that each event may trigger subsequent events. The counting process is shown in the middle panel of Figure 2.4.1 in the form of a nondecreasing step function, which is intuitive for the counts of events over time. The bottom panel shows how the arrival of an event will trigger the intensity to jump by a certain degree, the shape of which depends on the form of the memory kernel assumed and the parameter values used. This said, before we specify the assumed form of the memory kernel, we shall first detail an archetypal

[1] from sample cascade 1 in Figure 1.3.2

self-exciting point process in Section 2.4.1, called the Hawkes process.

2.4.1 Hawkes Process

The *Hawkes process* was introduced by Hawkes (1971) and serves as a natural way to model events where self-excitation is present. The process has found a wide variety of applications over the years, from its early use in seismology (Ogata, 1988) to its contemporary use in sociology (Rizoiu et al., 2017). Surveys with fine-grained details focusing on Hawkes processes and their applications in various fields have also been conducted, for example by Liniger (2009) and Zhu (2013).

The conditional intensity for the Hawkes process takes the following specific form

$$\lambda(t) = \nu + \sum_{i=1}^{N(t-)} \phi(t - \tau_i) = \nu + \int_{(0,t)} \phi(t - s) N(\mathrm{d}s), \tag{2.4.1}$$

where ν denotes the constant *baseline intensity* or the background rate, and $\phi(\cdot)$ is a function which governs the excitation effect or clustering density for each point τ_i, called the *memory kernel function*. The memory kernel function is used to account for the human response time on the social networks (Zhao et al., 2015; Kobayashi and Lambiotte, 2016), and has also been referred to as the delay function in some instances (Simma and Jordan, 2012).

To adapt to the various needs when modelling real-life phenomena, some more generalized versions of the process in (2.4.1), such as the one with time-varying baseline intensity (Chen and Hall, 2013) or different excitation functions (Mehrdad and Zhu, 2014) can prove to be useful. Moreover, the Hawkes process has also been studied in different contexts, giving rise to the so called intensity-based and cluster-based variants (Dassios et al., 2013). The form of the intensity function, together with the description of the cluster process interpretation, shall be presented in the following nested sections.

2.4.1.1 The Intensity Function

The baseline intensity ν in (2.4.1) describes the arrivals of events perturbed by exogenous interventions, which are referred to as the exogenous events or immigrants. It can be observed based on the intensity specified that the arrivals of these immigrants are independent of previous instances. However, the baseline intensity needs not take a constant value, but can be time-varying, which implies that the equation in (2.4.1) can be rewritten as

$$\lambda(t) = \nu(t) + \sum_{i=1}^{N(t-)} \phi(t - \tau_i). \tag{2.4.2}$$

Although many of the existing literature assumes self-exciting models with deterministic baseline intensities, such assumption seems to be unrealistic under many circumstances. An indicative example would be the work of Utsu (1961) which models the aftershocks of earthquake events. They discovered that the background aftershock rate shows a clear sign of temporal decay, which opines that the baseline intensity should decrease with time. Similarly, when modelling intraday stock trading, a self-exciting point process with a constant baseline intensity tends to fail when the intensity of trades is much higher during the opening of market compared to when the market is closing down (Engle and Lunde, 2003). Therefore, self-exciting point processes with varying baseline intensities often serve as more viable alternatives to model many real-life phenomena.

The memory kernel function $\phi(t - \tau_i)$, on the other hand, is responsible for accumulating the excitation effects up to time t for all the instances of $\tau_i \leqslant t$, which will jointly contribute to the event intensity at time t. The memory kernel is typically a monotonically decreasing function, so that more recent events will exert greater influence on the resultant intensity compared to events further away in the past. The form of the memory kernel assumed at the bottom panel of Figure 2.4.1 takes the following exponential decay form,

$$\phi(t) = \delta_1 e^{-\delta_2 t}, \tag{2.4.3}$$

for $\delta_1 \geqslant 0$, $\delta_2 > 0$, and $\delta_1 < \delta_2$. Specifically, the parameter δ_1 is used to denote the intensity jump right after the occurrence of an event, and the parameter δ_2 is used to account for the exponential decay. Both the parameters will jointly determine the clustering properties of the process, and it is usually the case that $\delta_1 < \delta_2$ to prevent the process from becoming explosive.

Based on the intensity in (2.4.1) where the memory kernel takes the form shown in (2.4.3), we used the parameter values $(\nu, \delta_1, \delta_2) = (0.5, 0.1, 0.5)$ to produce the pictorial output in the lower panel of Figure 2.4.1. This is done for demonstrative purposes, but the impacts of changing the parameter values, in particular the exponential decay parameter δ_2, should be noted. Figure 2.4.2 shows the strengths of excitation based on different parameter values of δ_2. Specifically, the top panel of Figure 2.4.2 uses $\delta_2 = 0.05$, a value much lesser than that of the δ_1 value. Consequently, the intensity tends to inflate continuously over time, making the process explosive. The middle panel of Figure 2.4.2 shows a more realistic decay at the parameter value of $\delta_2 = 0.5$, which is the value used in Figure 2.4.1 to produce a visually more sensible curve. Lastly, the bottom panel of Figure 2.4.2 shows a very fast decay at $\delta_2 = 5$, a value much greater than that of δ_1. Thus, it is important to have a set of sensible parameter values, as it tends to affect the predictions of future

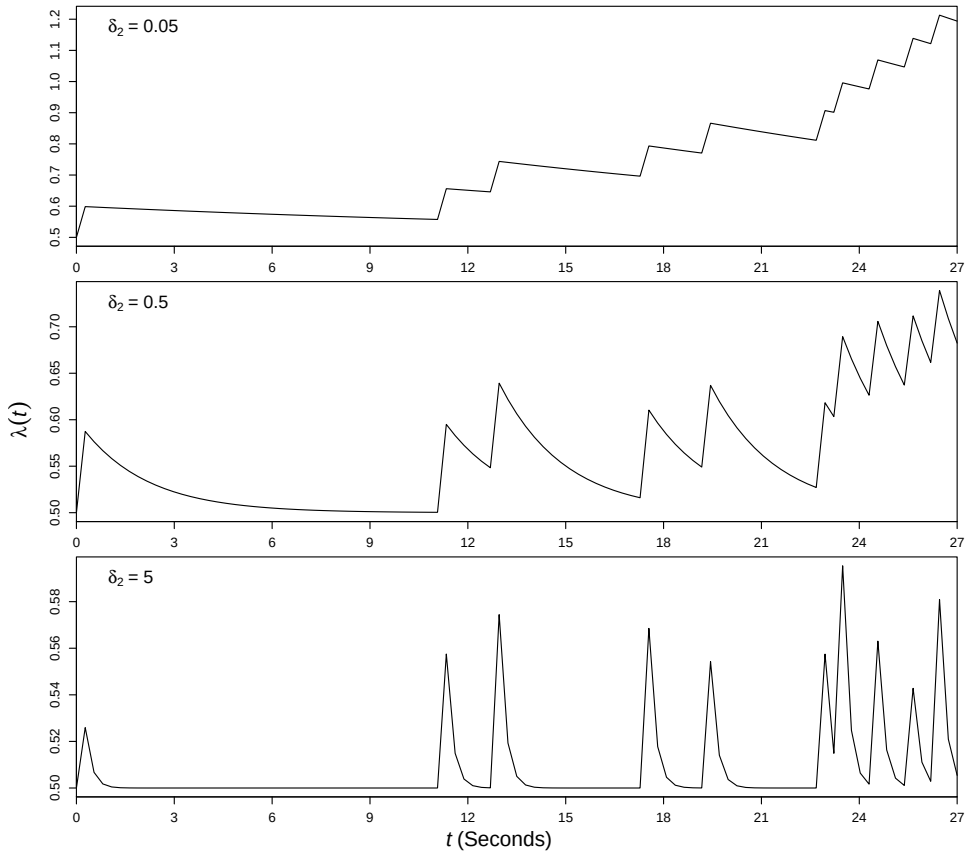

Figure 2.4.2: The effects of varying the exponential decay parameter values δ_2 at 0.05, 0.5, and 5 respectively from the top to the bottom panel, using the kernel in (2.4.3) and the form of intensity in (2.4.1). The baseline parameter is fixed at $\nu = 0.5$ and the jump size parameter is fixed at $\delta_1 = 0.1$. A suitable parameter value of δ_2 is required to produce a realistic decay over time.

events beyond the observation time.

The exponential kernel is typically the primary choice of kernel used in Hawkes processes, notably in financial data analysis (Filimonov and Sornette, 2015). As for the modelling of human dynamics on the social networks, the two main kernels used are the power-law kernel (Crane and Sornette, 2008) and the lognormal kernel (Zaman et al., 2014), the applicability of which depends on the platform of analysis.

2.4.1.2 Cluster Process Interpretation

When the intensity function $\lambda(\cdot)$ is linear, the Hawkes process is said to be linear and can be studied via the immigration-birth representation and interpreted as a Poisson cluster process (Hawkes and Oakes, 1974). This essentially categorizes the occurrences of events into *immigrants* and *offspring*, where the baseline intensity is responsible for generating the immigrants, and the memory kernel is responsible for generating the offspring. Specifically, immigrants will arrive independently to generate their respective offspring, thus forming their respective clusters. Such be-

haviour is attributable to the distinctive feature of the process, called the branching structure.

Recall from Figure 2.4.1 that we have events $\tau_1 < \tau_2 < \ldots < \tau_9$ originating from $\tau^0 = 0$. Suppose now that τ_1, τ_6 and τ_9 are immigrants. We assume that immigrant τ_1 directly generates τ_2 and τ_3, immigrant τ_6 directly generates τ_7, and immigrant τ_9 has no offspring. Furthermore, we assume that τ_3 generates both τ_4 and τ_5, whereas τ_7 generates τ_8. The branching structure forming clusters of events can be seen in Figure 2.4.3. The collection of immigrants $\{\tau_1, \tau_6, \tau_9\}$ can be

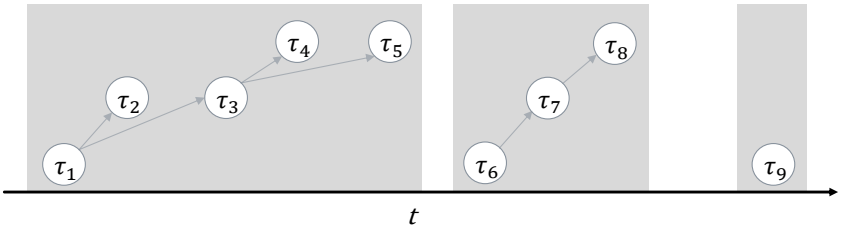

Figure 2.4.3: The branching structure of sample events $\tau_1 < \tau_2 < \ldots < \tau_9$. The shaded regions represent the individual clusters originating from immigrants. The first cluster consists of the events $\{\tau_1, \tau_2, \tau_3, \tau_4, \tau_5\}$, the second cluster consists of the events $\{\tau_6, \tau_7, \tau_8\}$, and the third cluster consists of the event $\{\tau_9\}$.

referred to as generation 0 events, and their direct offspring $\{\tau_2, \tau_3, \tau_7\}$ are called generation 1 events. Following this, the children of generation 1 events $\{\tau_4, \tau_5, \tau_8\}$ are called generation 2 events. Naturally, events of further generations adhere to such nomenclature. From Figure 2.4.3, it is clear that the set of events $\{\tau_1, \tau_2, \tau_3, \tau_4, \tau_5\}$ forms a cluster, $\{\tau_6, \tau_7, \tau_8\}$ forming another, and $\{\tau_9\}$ is also a cluster. This cluster process interpretation has important implications to simulate the process through efficient algorithms such as the cascading algorithm of Chen and Hall (2013, 2016), as we shall see later in Section 4.4.3.

One prime quantity associated with the branching structure of a Hawkes process is the *branching ratio/factor*, typically denoted by n^*, which corresponds to the expected number of events directly generated by an immigrant. By the memory kernel function in (2.4.3), the branching ratio can be conveniently expressed as

$$n^* = \int_0^\infty \phi(s)\,\mathrm{d}s = \int_0^\infty \delta_1 e^{-\delta_2 s}\,\mathrm{d}s = \frac{\delta_1}{\delta_2}. \tag{2.4.4}$$

As mentioned in Section 2.4.1.1, when the memory kernel takes the form in (2.4.3), it is necessary that $\delta_1 < \delta_2$ to prevent the process from becoming explosive. This equates to satisfying $n^* < 1$ in (2.4.4), a condition known as the *subcritical regime*. On the contrary, when $n^* > 1$, the process is said to be in the *supercritical regime*, meaning that the process tends to be explosive and is expected to generate an infinite or unbounded number of events. These regimes, when applied on different memory

kernel functions, can be useful in deducing if a point process can be numerically predicted.

2.5 Parameter Estimation

One of the most important procedures needed to understand the dynamics of a point process model is to estimate its parameters from the observed dynamics, frequently achieved by using the *maximum likelihood (ML)* approach.

The statistical problem we are considering involves determining the parameters for a conditional intensity function $\lambda(\cdot)$, which will then determine the distribution of the process N. Taking the intensity specified in (2.4.2), this asserts that we are considering a parametric problem where $\nu(\cdot)$ and $\phi(\cdot)$ are known up to a finite-dimensional parameter θ. In this particular example, the intensity of N satisfies,

$$\lambda(t;\theta) = \nu(t;\theta) + \sum_{i=1}^{N(t-)} \phi(t - \tau_i;\theta),$$

where $\theta \in \Theta \subset \mathbb{R}^d$, with Θ denoting the space of parameters and d is the dimension of the parameter vector. By the point process theory (Daley and Vere-Jones, 2003, Proposition 7.3.III), the likelihood of a point process with realizations at times $\{\tau_1, \tau_2, \ldots, \tau_{N(T)}\}$ over the interval of $[0, T]$, where T denotes the censoring time, takes the form

$$L(\theta) = \left\{ \prod_{i=1}^{N(T)} \lambda(\tau_i;\theta) \right\} \exp\left(- \int_0^T \lambda(t;\theta)\, \mathrm{d}t \right). \tag{2.5.1}$$

For computational convenience, the logarithm of the likelihood, or the log-likelihood,

$$\ell(\theta) = \log L(\theta) = \sum_{i=1}^{N(T)} \log \lambda(\tau_i;\theta) - \int_0^T \lambda(t;\theta)\, \mathrm{d}t, \tag{2.5.2}$$

is often used when obtaining the ML estimator $\hat{\theta}$ in practice. The maximization of the log-likelihood in (2.5.2) can be achieved by using various optimization techniques, such as the simplex search method of Nelder and Mead (1965) or those based on the Newton methods, discussed for example in Section 3.1 and Section 3.2 of Fletcher (2013).

2.6 Goodness-of-Fit Assessment

The assessment of the goodness-of-fit for a model, or the model adequacy, is often of interest after fitting the model to a data set. The residual point process approach

based on *Papangelou's random time change theorem* (Daley and Vere-Jones, 2003, Theorem 7.4.I) can be used in achieving this purpose. Specifically, if the collection of random points $\{\tau_i\}_{i=1,2,\ldots}$ follows the specified conditional intensity function over the interval $[0, T]$, then the integral transformed point pattern $\Lambda(\tau_i)$ should follow a unit rate Poisson process on $[0, \Lambda(T)]$, where $\Lambda(t) = \int_0^t \lambda(s)\,\mathrm{d}s$ is the cumulative intensity process.

Given the event times $\tau_1 < \tau_2, \ldots < \tau_{N(T)}$ up to the censoring time T, we can obtain the estimated parameter values $\hat{\theta}$ using the procedures discussed in Section 2.5. Since the joint distribution of the ordered event times would be equal to the order statistics of an equal number of uniformly distributed times over the interval $[0, T]$, the model adequacy can be inspected based on the uniformity of the transformed event times $\hat{\Lambda}(\tau_i)$ over the interval $[0, \hat{\Lambda}(T)]$, where

$$\hat{\Lambda}(\tau_i) = \int_0^{\tau_i} \hat{\lambda}(s)\,\mathrm{d}s,$$

for $\hat{\lambda}(\cdot) \equiv \lambda(\cdot; \hat{\theta})$. The uniformity of the residuals $\hat{\Lambda}(\tau_i)$ can be visually checked using the histogram or the quantile-quantile plots, or more formally through tests like the *Kolmogorov-Smirnov test*. When using the test of uniformity, we note that a larger p-value would indicate a better model fit. However, since the transformation function $\hat{\Lambda}$ carries some randomness in the observed data, the distribution of the test statistic would be more dispersed than that calculated from a stipulated Λ, which implies that smaller p-values should be tolerated.

2.7 Simulation of the Poisson Processes

To simulate a homogeneous Poisson process, we first note that the interevent times of the process are exponentially distributed. These exponential random quantities can be generated using the *inversion sampling* method, through sampling from $u \sim U(0, 1)$ and obtain an interevent time from the inverse cumulative distribution function $F^{-1}(u) = -\ln(u)/\lambda$, for a constant arrival rate λ. These exponential random variables can then be summed up to the target time point $\hat{T}$ to obtain a homogeneous Poisson process.

Simulating from an inhomogeneous Poisson process is slightly more challenging, and depends on the form of the intensity assumed. This said, we shall focus on simulation by thinning, similar to that proposed by Lewis and Shedler (1979) and Ogata (1981). The thinning property of the Poisson process postulates that the process intensity is piecewise constant such that it can be split into several independent processes, implying that an inhomogeneous Poisson process can be simulated through thinning its homogeneous counterpart with the intensity $\lambda_{max} \geqslant \lambda(\cdot)$. Thus, we

show below an efficient algorithm to simulate the inhomogeneous Poisson process given its intensity over the interval $[0, \hat{T}]$, following the `simPois` function in the `IHSEP` R package of Chen and Hall (2013, 2016):

1. Find the maximum intensity value from the input intensity over the interval $[0, \hat{T}]$ and denote it as λ_{max}.
2. Generate around $\lambda_{max}\hat{T} + 1.96\sqrt{\lambda_{max}\hat{T}}$ exponential variables based on λ_{max}.
3. Sum up all the generated exponential variables, and if their sum has not reached $\hat{T}$, iteratively generate more exponential variables in blocks with rate λ_{max}. The block size can be around $\sqrt{\lambda_{max}\hat{T}}$, but a size limit can also be set for efficiency.
4. Retain the cumulative sums of exponentials that are not more than $\hat{T}$ as the event times of the homogeneous Poisson process.
5. Perform thinning based on the retention probability $\lambda(t)/\lambda_{max}$ from the generated event times to obtain the event times for the inhomogeneous Poisson process.

Such efficient simulation method can be a building block to simulate more complex processes, such as the marked self-exciting point process detailed in Section 4.4.3, which is in turn useful for making predictions from those processes.

2.8 Prediction of Future Events

Various models can be built based on the theory of point processes, and by observing the dynamics up to the censoring time T, the predictions of future events up to a certain time point $\tilde{T}$ can be made. This is demonstrated as follows,

$$N(\tilde{T})_{\text{pred}} = N(T) + (N(\tilde{T}) - N(T))_{\text{pred}}, \tag{2.8.1}$$

where $N(T)$ is the observed number of events at time T, and $(N(\tilde{T}) - N(T))_{\text{pred}}$ is the predicted number of events from T to $\tilde{T}$.

For a relatively simple model like the Poisson process model, $N(\tilde{T}) - N(T)$ is Poisson distributed with its mean arrival rate equals to the integral of the identified intensity function from T to $\tilde{T}$, or

$$\int_T^{\tilde{T}} \hat{\lambda}(s)\,\mathrm{d}s \equiv \int_T^{\tilde{T}} \lambda(s;\hat{\theta})\,\mathrm{d}s, \tag{2.8.2}$$

where $\hat{\theta}$ is the estimated parameter values obtainable from various optimization routines. The mean rate in (2.8.2) can then serve as a point prediction.

For more complex processes such as the Hawkes process with time-varying baseline intensity, it is useful to consider the auxiliary point process $\tilde{N}(t) = N(T+t) - N(T)$ whose intensity process is given by,

$$\tilde{\lambda}(t) = \lambda(T+t) = \nu(T+t) + \sum_{i=1}^{N(T+t-)} \phi(T+t-\tau_i), \tag{2.8.3}$$

where we note that $N(\tilde{T}) - N(T) = \tilde{N}(\tilde{T} - T)$. Using (2.8.3), the predicted numbers of future events, in the forms of the conditional expectation and the conditional median over the interval $(0, \tilde{T} - T]$ can be obtained through simulation-based approaches. However, if only the conditional expectation is needed, we can construct an integral equation based on (2.8.3) and solve it numerically to obtain the mean intensity on $[0, \tilde{T} - T]$, using for example some flexible parametric functions to approximate the unknown function in the integral equation.

Chapter 3

Existing Prediction Methods

The phenomenal influence exerted by Twitter has motivated a multitude of research over the past few years, ranging from deriving factors which make a tweet more popular than others to observing the retweeting dynamics and predicting the popularity of tweets based on such dynamics. As per our previous discussion, the popularity of a tweet is conventionally measured by the total number of retweets generated by an original tweet up to a certain time point.

Numerous prediction methods have been proposed to forecast the future popularity of tweets. These methods can require relatively simple inputs like the retweet times and the corresponding numbers of followers of the retweeters, or more extensive features such as the complete network structure or users' demographic profiles. Despite many of the existing prediction methods proposed focus specifically on Twitter, methods applied on different social media platforms are also noteworthy for their potential utility in tweet popularity predictions.

This chapter revolves around describing existing works on popularity predictions, with special emphasis on the point process models of Zhao et al. (2015) and Kobayashi and Lambiotte (2016), namely the SEISMIC and the TiDeH model, as they were found to have outstanding prediction performances. The remainder of this chapter is organized as follows. We first give an overview of popularity predictions, highlighting their various applications and distinguishing the different classes of existing prediction methods in Section 3.1. The specifics of these methods in terms of the approaches employed and the various limitations prevalent are subsequently presented in Section 3.2. We proceed to exhibiting the forms assumed by the SEISMIC and the TiDeH model, and discuss how these models can perform tweet popularity predictions in Section 3.3 and Section 3.4 respectively. Lastly, we demonstrate how the performances of different prediction methods can be assessed, using the evaluation metrics suggested in Section 3.5.

3.1 Overview

Accurate popularity predictions can prove valuable to actors playing different roles on the internet. Public users can avoid the problem of information overload as only the most relevant information will be displayed, content providers can better organize the information to make the platform more user-friendly, and advertising firms can design more profitable strategies to earn additional profits. This correlates to the wide applicability of popularity predictions, notably in web distribution systems and online marketing (Tatar et al., 2014). While we elaborate herein the usefulness of popularity predictions on a broader perspective encompassing the different varieties of web contents, it should be intuitive that these contents may refer specifically to tweets.

The use of popularity predictions to cache and replicate contents more efficiently has been emphasized in our preliminary discussion of popularity prediction in Section 1.2.2. This said, reliable forecasts of popularity can also be of immense help in online marketing (Lakkaraju and Ajmera, 2011), and are essential in setting up powerful recommendation systems. By recommending likeable items to the right audience, the user experience can be enhanced, thereby boosting the site's traffic and attracting more revenues. Therefore, advertising agencies can use such predictive capability to develop strategies for online advertisements (Wu and Shen, 2015) to earn additional profits and promote their reputation in a more dramatic fashion. Popularity prediction can also be used to forecast various real-world outcomes, such as candidates most likely to achieve electoral success. This brings the attention to Twitter, which has remained one of the most influential social media platforms to disseminate election-related information (Isaac and Ember, 2016).

Popularity prediction methods can be categorized based on the granularity of information used in the prediction process (Tatar et al., 2014), as depicted in Figure 3.1.1, where we note that the term domain refers to the venue wherein a web content resides. Based on Figure 3.1.1, the information used in the prediction process can come from the local domain, or from a different domain, known as cross domain. When modelling the information diffusion on Twitter, this equates to saying that the dynamics are observed internal or external to the network. The idea of cross domain prediction has been supported by the interconnectedness of social media which has endowed Twitter with the ability to share contents from various external sources such as online video sites, online news sites, social bookmarking sites, and other social networking sites.

One of the most challenging tasks when performing a local domain prediction is to predict the popularity prior to the publication of the content, or *pre-publication popularity prediction*. Such prediction is usually achieved by relying on the metadata

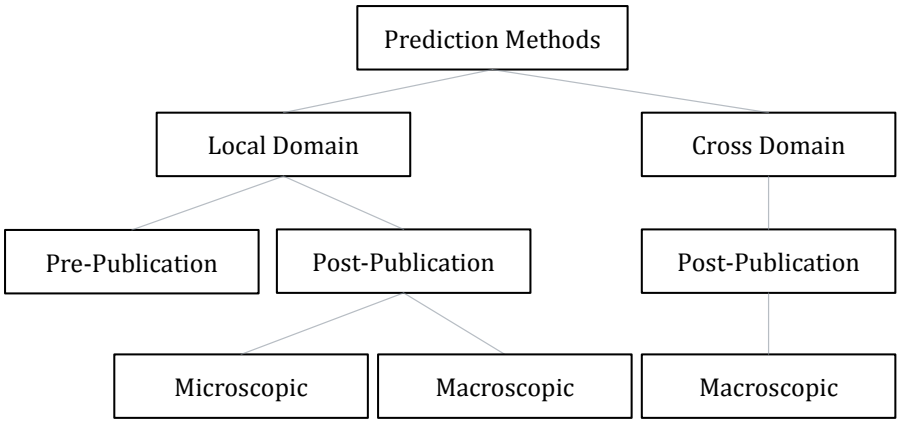

Figure 3.1.1: Prediction methods based on the granularity of information used. The methods can be based on local or cross domain, using information internal or external to the platform. Pre-publication and post-publication methods refer respectively to predictions made before and after observing events. Predictions at the microscopic and macroscopic levels correspond to how the predictions are made using individual and aggregated data.

or features such as the social connections of publishers. Pre-publication prediction on Twitter, specifically, refers to how prediction can be made based solely on the information readily available at time zero. For instance, one might question how many retweets will his tweet attract up to a certain time, given features like his number of followers and the intended publication time, both of which are available in the Twitter data set discussed in Section 1.3. Furthermore, information like the tweeter's geographical location or the semantic features of the tweet can prove to be useful in obtaining a more accurate popularity prediction, although it requires more extensive feature engineering.

Most prediction methods in the literature need to rely on some observations prior to making any predictions, so called the *post-publication popularity predictions*. Under such prediction methods, the modelling procedures can be implemented either at a *microscopic* or *macroscopic* level, where the former refers to modelling at an individual user's level, and the latter at an aggregated level. Microscopic level methods leverage the dynamics of heterogeneous users by giving them unique treatment, with many of the existing prediction methods falling into this category. Macroscopic level methods, in comparison, deduce the future popularity by assuming that the users are relatively similar to each other, or are homogeneous.

Considering how users' actions are not constrained on a single platform, that is, some users may exhibit certain patterns of behaviours across different platforms, cross domain popularity prediction methods can come in handy. It is often the case that Twitter users tend to have multiple accounts spanning across different social media platforms, which makes explaining popularity from the perspective of a single

domain insufficient. A natural approach to overcome this problem is to extract and transfer information across the platforms. However, as it is nearly impossible to identify a user across two different platforms, such method is only known to have been constructed at a macroscopic level. Additionally, faced with the difficulties and practicality concerns when deducing cross-domain behaviours, methods under this class are expectedly scarce.

3.2 The Specifics

Having conferred an overview of popularity predictions in Section 3.1, we shall present the method specifics in this section. An abundance of popularity prediction methods have been proposed in the last decade, ranging from simple linear regression functions to complicated frameworks which correlate information across websites.

3.2.1 Local Domain

As mentioned in Section 3.1, most methods in the literature are local domain prediction methods where predictions can be made before or after the content publications, known respectively as pre-publication and post-publication prediction methods.

Pre-publication prediction methods are useful for predicting the popularity of web contents with relatively short lifespans, which apply to the majority of tweets shared by non-elite users on the Twitter network. Existing works that specifically address the problem of pre-publication tweet popularity prediction have been scarce, but one of which with an intriguing discovery is that conducted by Martin et al. (2016). In particular, they proposed a model which distinguishes between two main sources of inaccuracy in pre-publication predictions, namely errors in the predictive model itself and the unpredictability of social system given its complexity. The model was then used to perform tweet popularity prediction by relying on a set of extensive features such as the prior knowledge on the popularity levels of tweets with nearly identical contents. Although the model prediction performance appears rather promising, they reported that the prediction accuracy can be further improved if the system in question is homogeneous and the prior knowledge is flawless.

Another noticeable work focusing on pre-publication prediction is that of Bandari et al. (2012), where the number of tweets was used as an indicative measure of news popularity. The authors formulated the prediction tasks as both a numerical and classification problem, and concluded that while predicting the popularity as a single numeric value, or point prediction, is prone to large errors, predicting the popularity in a range, or interval prediction, shows outstanding prediction accuracy. However, the popularity of news articles here calculates the number of tweets shared with

the specific content, rather than the conventional number of retweets stemming from a single tweet. This asserts that the popularity may be affected more by the interestingness and degree of exposition of the content, instead of the distinctive users' features on the Twitter network.

Most, if not all, of the popularity prediction methodologies in the literature assume observations of the retweet sequence of a tweet for a period of time before a prediction can be made. Such prediction methods, as discussed in Section 3.1, are referred to as post-publication popularity prediction methods, and can be broadly classified into microscopic and macroscopic levels, with the prerequisites being the individual retweeting dynamics and aggregated popularity over time respectively.

3.2.1.1 Microscopic Level Methods

A microscopic level method draws conclusions based on individual user's behaviour, and is mostly constructed by fitting a point process model to the observed retweet sequence up to the censoring time, and then projecting the fitted point process to a future time point. Examples of methods under this level include those based on the Self-Exciting Model of Information Cascades (SEISMIC) of Zhao et al. (2015), the Time-Dependent Hawkes (TiDeH) model of Kobayashi and Lambiotte (2016), and the multilevel model of Zaman et al. (2014).

The SEISMIC of Zhao et al. (2015) describes the retweet intensity of a tweet, or the expected number of retweets per unit time, as a product of the infectivity of the original tweet and the accumulated excitation effects of all previous retweets. Zhao et al. (2015) estimated the infectivity as a function of time using a kernel smoothing estimator, and the excitation function, or the memory kernel, using a graphical approach under the assumption that some retweeting processes follow an inhomogeneous Poisson process with the excitation function as its intensity function. They also proposed to predict the future popularity of a tweet based on calculating the expected number of future retweets by first assuming that the infectivity remains constant since the censoring time, and a subsequent ad hoc adjustment to the expectation to incorporate the decaying trend of the infectivity. They reported that the predictions of tweet popularity using their approach outperform those based on some competing approaches (Crane and Sornette, 2008; Agarwal et al., 2009; Szabo and Huberman, 2010; Gao et al., 2015), under several performance measures.

The TiDeH model of Kobayashi and Lambiotte (2016) models the retweet intensity similar to the SEISMIC. They first estimated the infectivity and memory kernel using similar nonparametric kernel smoothing estimators. Then, they fitted a circadian rhythmic function to the nonparametrically estimated infectivity function up to the censoring time, and extrapolated it beyond the censoring time to predict the future number of retweets. With certain choices of the smoothing pa-

rameters, the tweet popularity predictions based on the TiDeH model are superior to those based on the SEISMIC, especially on longer cascades. However, Kobayashi and Lambiotte's approach requires sufficiently long observation time on a retweet sequence to have reliable estimation of the infectivity function, and the prediction performance depends critically on the window size parameter used in the estimation step of the infectivity function.

Another noticeable method under the microscopic level is that of Zaman et al. (2014), where a multilevel model based on the branching process was used together with a Markov Chain Monte Carlo (MCMC) Bayesian approach for inference. The probabilistic model was built based on the assumption that the pool of Twitter users portrays similar behaviours when reacting to a tweet, which then generates distinguishable patterns in the evolution of the tweet popularity. The model was able to produce reliable forecasts by observing from the retweet cascades within a matter of minutes, but the results were only testified on a very small data set. Moreover, the method requires extra knowledge about the network structure among the original tweeter and the retweeters, which, unfortunately, is not available in the data set described in Section 1.3.

3.2.1.2 Macroscopic Level Methods

A macroscopic level method predicts the popularity based on aggregated users' attention, and is often a faster alternative to obtain the popularity estimate. The methods under this level can usually be further classified into studies focusing on aspects such as the cumulative growth of popularity and temporal analysis (Tatar et al., 2014). The former reveals the popularity level of a web item since its publication to the prediction time point. The latter shows how the popularity temporally evolves up to the prediction time, with the time element being a prerequisite.

A logistic regression model was proposed by Hong et al. (2011) to predict the future popularity of a tweet based on the cumulative growth of popularity. They addressed the prediction problem as a classification task consisting of several classes, and aimed specifically to predict the range of future popularity. By using a logistic regression classification function which makes use of topological, temporal, and content features, they reported that the proposed methodology can uncover which tweets will not receive any retweets at all, and which tweets tend to receive myriads of retweets. A more recent model proposed by Lymperopoulos (2016), along similar lines, captures the cumulative growth of popularity by interlacing linear and nonlinear growth terms, which correspond respectively to stationary and nonstationary adoption phases. The model demonstrates a great fit to the empirical popularity patterns, and is able to generate accurate forecasts of future popularity via extrapolation.

From the perspective of temporal analysis, Kong et al. (2014) proposed a model motivated by the k-nearest neighbour algorithm to predict the tweet popularity a certain number of days after its publication, based on the dynamics observed in the first hour. Specifically, when a new tweet is detected, the algorithm calculates the similarity index between the tweet and all other historical tweets published by the same user, identifies the top-k most similar tweets based on the features extracted from the time series of the retweet sequences, and estimates the popularity based on the average popularity of these identified tweets. Kong et al. (2014) concluded that their method outperforms a number of regression-based methods by a considerable margin, although the implementation can be computationally demanding.

Before we proceed to describing models which cross-correlate information from different platforms, we shall highlight the hybrid model proposed by Mishra et al. (2016), which combines both the microscopic and macroscopic level prediction methods. In particular, the hybrid model makes use of a microscopic level predictor based on a marked Hawkes process, and a macroscopic level predictor based on a set of extensive features. The microscopic component of the method uses a power-law memory kernel to model the retweeting dynamics, and a cluster process interpretation to predict the future number of events. This is then combined with the macroscopic component which uses certain key features, such as basic user features and temporal features, to improve the prediction performance. The hybrid model bridges the gap of some undesirable limitations extant when employing a standalone generative or feature-driven approach, and is especially useful when extra features can be observed.

3.2.2 Cross Domain

We have discussed in Section 3.1 that the contents shared on Twitter can come from various external sources such as online video sites, online news sites, social bookmarking sites, and other social networking sites. A natural question to ask is whether the popularity of a web item on one of these platforms can reflect its popularity on Twitter, and contrariwise, if the tweet popularity can be predictive of the popularity of a similar web item in an external environment. As compelling as it may seem, popularity prediction which cross-correlates information across different platforms, or cross-domain popularity prediction, has remained largely unexplored.

Due to the difficulties in proper identifications of users across different platforms, approaches in the literature seem to have only included macroscopic level methods, with emphasis on post-publication popularity predictions. The algorithm proposed by Roy et al. (2013), for example, extracts information from Twitter to

detect videos likely to experience sudden bursts of popularity on Youtube[1]. Bursty videos are lucrative to detect, since the sudden rise in popularity of such videos provides a unique opportunity for advertising and caching. The procedures involved in cross-correlating the information is rather straightforward. That is, hot topics on Twitter are extracted and associated with videos on Youtube, both of which are then compared in terms of their popularity levels in their respective domains, and if the popularity of a topic on Twitter overwhelms that on Youtube, it is a clear indication that the video is susceptible to a sudden burst of popularity on Youtube. It should be noted, however, that the popularity of a Youtube video is defined differently here, by the total number of views the video has attracted up to a certain time point.

Note how the relationship demonstrated in Roy et al.'s work is unidirectional, that is, a hot topic on Twitter can speak volume on the popularity of a video on Youtube, but a viral Youtube video might not necessarily be popular on Twitter. This can be attributed to the fact that Youtube is an open channel comprising of videos viewable by the general public, whilst videos shared on Twitter are visible only to followers or viewers visiting the profile page. More importantly, if a Youtube video is shared by a non-elite user on Twitter who has a small number of followers, then the video will most likely be not getting much attention at all, at least on the tweeter's profile page. This signals that the tweet popularity depends largely on the tweeter's features, and that its prediction based on external media is usually infeasible.

As a remark, the work of Oghina et al. (2012) is also noteworthy for its capability to make cross-domain prediction, where the ratios of likes and dislikes on Youtube is combined with the positive and negative unigrams on Twitter to predict the ratings of movies on IMDb[2]. A survey on popularity prediction methods from a slightly different perspective can be found in Li et al. (2017). Next, we shall explain in details the prediction approaches proposed by Zhao et al. (2015) and Kobayashi and Lambiotte (2016), in Section 3.3 and Section 3.4 respectively.

3.3 The Self-Exciting Model of Information Cascades

The intensity function of the SEISMIC (Zhao et al., 2015) which captures the dynamics of the retweeting process is given by

$$\lambda(t) = p(t) \sum_{i=0}^{N(t-)} n_i \phi(t - \tau_i), \quad t > 0, \tag{3.3.1}$$

[1]https://www.youtube.com/
[2]https://www.imdb.com/

where $n_i, i = 0, 1, \ldots$ denotes the i.i.d event mark or the number of followers, and $\tau_i, i = 0, 1, \ldots$ denotes the event time, or the time of tweet at $i = 0$ or retweet at $i = 1, 2, \ldots$. As per our discussion in Section 2.4.1, the function $\phi(\cdot)$ is referred to as the *memory kernel function*, and accounts for the human response time. The extra component function $p(\cdot)$, on the other hand, is known as the *infectivity function*, and accounts for the tweet virality.

To estimate the memory kernel function $\phi(\cdot)$, Zhao et al. (2015) selected 15 relatively popular tweets in the training data set to approximate the parameters and used these parameters as a part of the intensity process in (3.3.1) to predict the future number of events. These tweets were assumed to have the following probability density function

$$\phi(s) = \begin{cases} c & \text{if } 0 < s \leqslant s_0 \\ c(\frac{s}{s_0})^{-(1+\beta)} & \text{if } s > s_0, \end{cases} \tag{3.3.2}$$

where $s_0 = 300$ seconds for a constant reaction time distribution, followed by a power-law decay afterwards. The parameter c in (3.3.2) is a constant which can be estimated by making use of the basic property of a probability density function,

$$\int_0^\infty \phi(s)\,\mathrm{d}s = c\int_0^{s_0} 1\,\mathrm{d}s + cs_0^{1+\beta}\int_{s_0}^\infty s^{-(1+\beta)}\,\mathrm{d}s = 1. \tag{3.3.3}$$

By fitting the aforementioned popular tweets, the parameters in (3.3.3) were found to be $\beta = 0.242$ and $c = 6.27 \times 10^{-4}$. The implementation of such an approach ensures that the SEISMIC can produce results in a computationally inexpensive way, as the estimation procedure would only involve $p(\cdot)$, which is defined nonparametrically as

$$p(t) = \frac{\sum_{i=1}^{N(t-)} K(t - \tau_i)}{\sum_{i=0}^{N(t-)} n_i \int_{\tau_i}^t K(t-s)\phi(s-\tau_i)\,\mathrm{d}s}. \tag{3.3.4}$$

The estimation in (3.3.4) can be performed by using a certain smoothing function, for example the triangular kernel function

$$K(s) = \max\left\{1 - \frac{2s}{t}, 0\right\}, \quad s > 0, \tag{3.3.5}$$

to discard posts as they get stale. Specifically, Zhao et al. (2015) required posts older than $t/2$ to be discarded by the kernel. Their choice of the triangular kernel was driven by its several desirable properties, such as its ability to discard unstable and potentially explosive period at the incipient phase of observation, its adjustable window size based on the time t, its emphasis on more recent posts compared to

older posts in the window, and its piecewise linear form which gives the integral $\int K(t-s)\phi(s-\tau_i)\,\mathrm{d}s$ a closed form expression.

After estimating the parameters, the SEISMIC can make predictions based on two important regimes, namely the supercritical and subcritical regimes. Specific to information diffusion modelling on Twitter based on the SEISMIC, we have

1. Supercritical regime: if $\hat{p}(T) \geqslant \frac{1}{R}$, then $\mathbb{E}\left[N(\tilde{T})\middle|\mathcal{F}_T\right] \to \infty$ as $\tilde{T} \to \infty$.

2. Subcritical regime: if $\hat{p}(T) < \frac{1}{R}$, then $\sup \mathbb{E}\left[N(\tilde{T})\middle|\mathcal{F}_T\right] < \infty$.

The parameter $R = \mathbb{E}[n_i]$ is the expected number of followers, referred to as the *expected response*, and can be estimated from the training data set. These two classes of regimes have been previously discussed in Section 2.4.1.2 when we touch on the branching structure of Hawkes processes. Furthermore, recall that $\mathcal{F}_T$ denotes the history of the retweeting process up to the censoring time T. As highlighted in Section 2.4.1.2, when a tweet is under the supercritical regime, it is considered explosive and is expected to generate an infinite number of retweets as the time extends to infinity. Thus, to make a sensible prediction, the SEISMIC needs to satisfy $\hat{p}(T) < 1/R$.

By the SEISMIC approach, the expected number of events from T to $\tilde{T}$ given the history up to time T, or $\mathbb{E}[N(\tilde{T}) - N(T)|\mathcal{F}_T]$, can be calculated based on the branching process interpretation,

$$\begin{aligned}\mathbb{E}\left[N(\tilde{T}) - N(T)\middle|\mathcal{F}_T\right] &= \mathbb{E}\left[\sum_{k=1}^{\infty} N^k\right] \\ &= \frac{\mathbb{E}[N^1]}{1 - R\hat{p}(T)} \\ &= \frac{\hat{p}(T)}{1 - R\hat{p}(T)} \sum_{i=0}^{N(T-)} n_i \left(1 - \int_{\tau_i}^{T} \phi(s-\tau_i)\,\mathrm{d}s\right),\end{aligned} \tag{3.3.6}$$

where N^k denotes the number of k^{th} generation events, and the functions $p(\cdot)$ and $\phi(\cdot)$ take the forms shown in (3.3.4) and (3.3.2) respectively. After obtaining (3.3.6), Zhao et al. (2015) proposed an impromptu adjustment to further improve the prediction accuracy, based on the incorporation of two scaling constants to the conditional expectation. Specifically, they added the constant κ to ensure that the infectivity will decay and eventually die out after a sufficiently long time, and the constant ψ to account for the possible mutuality in the followers of retweeters. Adapting such constants to (3.3.6) yields a slightly different conditional expectation of the number

of events,

$$\mathbb{E}\left[N(\tilde{T}) - N(T)\middle|\,\mathcal{F}_T\right] = \frac{\kappa\hat{p}(T)}{1-\psi R\hat{p}(T)}\sum_{i=0}^{N(T-)} n_i\left(1-\int_{\tau_i}^{T}\phi(s-\tau_i)\,\mathrm{d}s\right). \quad (3.3.7)$$

These scaling constants were naturally estimated from the training data set, based on minimizing the median absolute percentage error. In addition, as the impact of κ on the prediction accuracy was found to be much more significant than that of ψ, Zhao et al. (2015) fixed the value of ψ but allowed κ to be time-varying.

From (3.3.7), we can predict the popularity of a tweet by using individual vector of times τ_i and marks n_i for $i = 0, 1, \ldots$ alongside with a specified censoring time T. This can be implemented conveniently using Zhao et al.'s R package `seismic`. In general, the computational cost of the SEISMIC is inexpensive, although the prediction accuracy may sometimes be lacklustre. The efficiency of the SEISMIC can be attributed to the closed forms assumed by the memory kernel in (3.3.2) and the triangular kernel in (3.3.5), both of which are piecewise-polynomials.

Despite the efficiency of the SEISMIC, its nonparametric form restrains the possibility of performing useful simulations. Furthermore, one major limitation of the SEISMIC is that it considers some tweets to be explosive and unpredictable, which may be a consequence of model misspecification. To cope with tweets portraying explosiveness, one can perform experiments using different memory kernel functions, or resort to using various alternative models, for instance the TiDeH model proposed by Kobayashi and Lambiotte (2016), discussed next in Section 3.4.

3.4 The Time-Dependent Hawkes Model

The Time-Dependent Hawkes (TiDeH) model of Kobayashi and Lambiotte (2016) is a variant of the SEISMIC useful in fitting and predicting the popularity of longer retweet cascades. The TiDeH model predicts the popularity of a tweet by summing up individual expected number of events in equally-spaced time windows, from a censoring time T to a certain prediction time $\tilde{T}$.

The forms of the intensity and memory kernel functions of the TiDeH model are identical to those of the SEISMIC in (3.3.1) and (3.3.2) respectively, but the infectivity function $p(\cdot)$ now involves a two-step approach estimation, albeit the similar assumption on its decay over time. In particular, both the SEISMIC and the TiDeH model assume a time-decreasing infectivity, which is intuitive as the virality, or newsworthiness of a tweet, should decay and eventually die out after a sufficiently long time. This implies that highly infectious tweets can last for weeks or even months before they get stale and lose interestingness completely, but tweets with considerably low infectivity can die out almost instantly after their publications.

The TiDeH model assumes both nonparametric and parametric forms of the infectivity functions as shown in (3.4.1) and (3.4.2),

$$p_0(t) = \frac{N(t_a, t_b)}{\sum_{i=0}^{N(t-)} n_i \left\{\Phi(t_b - \tau_i) - \Phi(t_a - \tau_i)\right\}}, \tag{3.4.1}$$

$$p(t) = \alpha_0 \exp\left(-\frac{t}{\beta_0}\right)\left\{1 - \gamma_0 \sin\left(\frac{2\pi}{T_d}\,(t + \delta_0)\right)\right\}. \tag{3.4.2}$$

We note from (3.4.1) that n_i denotes the number of followers of the i^{th} retweeter, the function $\Phi(\cdot)$ is the integral of the memory kernel in (3.3.2) where $\Phi(t) = \int_0^t \phi(s)\,\mathrm{d}s$, and $[t_a, t_b]$ is a moving time window from which t falls in, with window size $\Delta = t_b - t_a$. The function (3.4.2) consists of the parameters $\alpha_0, \beta_0, \gamma_0$, and δ_0 to account for the retweet intensity, the characteristic time of popularity decay, the relative amplitude of oscillation, and its phase. In addition, the parameter T_d used to denote the oscillation period is naturally fixed at one day to reflect the diurnal patterns of activity levels.

The nonparametric function in (3.4.1) is essentially the preliminary estimate of the tweet infectivity, followed by its subsequent parametric estimation in (3.4.2) which adapts a time-dependent oscillating function to account for the repetitiveness of human routine activities. To estimate the parameters, Kobayashi and Lambiotte (2016) proposed to minimize the sum of squares of $p_0(k) - p\left((k + 0.5)\Delta\right)$ over all the time windows. They calibrated the TiDeH model by using all the extremely popular tweets, each with at least 2,000 retweets, available in the whole data set described in Section 1.3.

With the fitted parameters $\hat{\alpha}_0, \hat{\beta}_0, \hat{\gamma}_0, \hat{\delta}_0$ for a retweet cascade obtained, its future evolution can be predicted. For that, it would be useful to express the intensity function of the TiDeH process beyond the censoring time T, denoted by $\tilde{\lambda}(t)$, shown as follows,

$$\begin{aligned}
\tilde{\lambda}(t) &= \lambda(T + t) \\
&= p(T + t) \sum_{j=0}^{N(T+t-)} n_j \phi(T + t - \tau_j) \\
&= p(T + t) \sum_{j=0}^{N(T)} n_j \phi(T + t - \tau_j) + p(T + t) \sum_{j=N(T)+1}^{N(T+t-)} n_j \phi(T + t - \tau_j) \\
&= \tilde{\nu}(t) + p(T + t) \sum_{j=1}^{\tilde{N}(t-)} n_{N(T)+j} \phi(t - (\tau_{N(T)+j} - T)) \\
&= \tilde{\nu}(t) + \tilde{p}(t) \sum_{j=1}^{\tilde{N}(t-)} \tilde{n}_j \tilde{\phi}(t - \tilde{\tau}_j).
\end{aligned} \tag{3.4.3}$$

The intensity in (3.4.3) is essentially the intensity process for $\tilde{N}(t) = N(T+t) - N(T)$, or a temporally shifted version of the intensity in (3.3.1) with the baseline intensity,

$$\tilde{\nu}(t) = p(T+t) \sum_{j=0}^{N(T)} n_j \phi(T + t - \tau_j).$$

To predict the future popularity, Kobayashi and Lambiotte (2016) proposed to use a method based on solving an integral equation, which requires a proper definition of the mean intensity function of the TiDeH process, denoted by $\bar{\lambda}(t)$, illustrated as follows,

$$\begin{aligned} \bar{\lambda}(t) &= \mathbb{E}\left[\tilde{\lambda}(t) \middle| \mathcal{F}_T\right] \\ &= \mathbb{E}\left[\tilde{\nu}(t) + \tilde{p}(t) \sum_{j=1}^{\tilde{N}(t-)} \tilde{n}_j \tilde{\phi}(t - \tilde{\tau}_j) \middle| \mathcal{F}_T\right] \\ &= \tilde{\nu}(t) + R\tilde{p}(t) \int_0^t \tilde{\phi}(t-\tau)\bar{\lambda}(\tau)\,\mathrm{d}\tau. \end{aligned} \tag{3.4.4}$$

Similar to the SEISMIC, $R = \mathbb{E}[n_i]$ is used to denote the expected response. It should be noted, however, that despite that same notation, they are calculated differently and are used in different scenarios. For the SEISMIC, R is used to determine the regime from which a cascade falls in, and is also used in the prediction process together with the scaling constant ψ, as indicated in (3.3.7). The value should, in principle, be estimated based solely on the numbers of followers in the training data set, but can be adjusted accordingly to a smaller volume to prevent too many instances of cascades falling under the supercritical regime. As for the TiDeH model, the value of R is estimated based on the numbers of followers of the tweeter and previous retweeters within the same retweet cascade, up to time T.

To obtain the conditional expectation of the number of events from T to $\tilde{T}$ given its history, or $\mathbb{E}[N(\tilde{T}) - N(T)|\mathcal{F}_T]$, the integral equation in (3.4.4) needs to be solved numerically, for example using the *B-spline function* with sufficiently many knots and a certain order. Specifically, let $B(t) = (B_1(t), B_2(t), \ldots, B_k(t))^\top$ denote the set of B-spline basis functions of a certain order on the interval $(0, \tilde{T} - T]$ with a further assumption that $\bar{\lambda}(t) \approx B(t)^\top \eta$ for a k-vector η. Then, by solving the equation for η as follows

$$B(t)^\top \eta = \tilde{\nu}(t) + \left\{ R\tilde{p}(t) \int_0^t \tilde{\phi}(t-\tau) B(\tau)^\top \eta \,\mathrm{d}\tau \right\}, \tag{3.4.5}$$

we can predict the number of events in $(0, \tilde{T} - T]$ using $(\int_0^{\tilde{T}-T} B(t)\,\mathrm{d}t)^\top \eta$.

Different from the SEISMIC, the TiDeH model has its intensity function taking a parametric form. This implies that the TiDeH $\tilde{N}$ process can be simulated by using

some appropriate algorithms, such as the *rejective method* of Lewis and Shedler (1979). The method involves serially generating events one after another, and for that purpose, we need to first define the hazard functions for the TiDeH $\tilde{N}(t)$ process based on its $\tilde{\lambda}(t)$ intensity. Specifically, the first event beyond the censoring time T can be generated based on the hazard function $h_{\tilde{\tau}_1}(t)$, the second event can be generated based on the hazard function $h_{\tilde{\tau}_2}(t)$, and by this convention the i^{th} event can be generated based on the hazard function $h_{\tilde{\tau}_i}(t)$. The equations in (3.4.6) shed light on the specific forms of the process intensities,

$$\begin{aligned}
h_{\tilde{\tau}_1}(t) &= \lambda(T+t) = p(T+t) \sum_{j=0}^{N(T)} n_j \phi(T+t-\tau_j), \\
h_{\tilde{\tau}_2}(t) &= \lambda(T+\tilde{\tau}_1+t) = p(T+\tilde{\tau}_1+t) \sum_{j=0}^{N(T)+1} n_j \phi(T+\tilde{\tau}_1+t-\tau_j), \\
h_{\tilde{\tau}_i}(t) &= \lambda(T+\tilde{\tau}_{i-1}+t) = p(T+\tilde{\tau}_{i-1}+t) \sum_{j=0}^{N(T)+i-1} n_j \phi(T+\tilde{\tau}_{i-1}+t-\tau_j).
\end{aligned} \tag{3.4.6}$$

Intuitively, the events $\tilde{\tau}_i$ will be generated from T to $\tilde{T}$, where it is important to take note of the relationship,

$$\tau_{N(T)+i} = T + \tilde{\tau}_i. \tag{3.4.7}$$

With the information on time and mark (τ_i, n_i) available for $i = 0, 1, \ldots, N(T)$ and the definitions of hazard functions in (3.4.6), events over the interval $(0, \tilde{T} - T]$ for the TiDeH $\tilde{N}(t)$ process can be simulated by using the following procedures,

1. Define the maximum intensity by $\tilde{\lambda}_m(t) = \max \tilde{\lambda}(t)$ where the initial max intensity is set to $\tilde{\lambda}_m(0)$ for $t = 0$.

2. Generate the first proposed event $\tilde{\tau}_1^*$ based on $Exp(\tilde{\lambda}_m(0))$.

3. Generate a number $V \sim U(0,1)$ and,

   - If $V \leqslant h_{\tilde{\tau}_1}(\tilde{\tau}_1^*)/\tilde{\lambda}_m(0)$ then $\tilde{\tau}_1^*$ is an event time, making $\tilde{\tau}_1^* \equiv \tilde{\tau}_1$. In this case, generate $\tilde{n}_1$ from previous marks n_i for $i = 0, 1, \ldots, N(T)$ and add the validated event time and mark $(\tau_{N(T)+1}, n_{N(T)+1})$ to the existing pairs of event times and marks $\{(\tau^0, n^0), (\tau_1, n_1), \ldots, (\tau_{N(T)}, n_{N(T)})\}$ following (3.4.7). Then, update the max intensity by $\tilde{\lambda}_m(\tilde{\tau}_1)$ and generate the next proposed event $\tilde{\tau}_2^*$ using $\tilde{\tau}_1 + Exp(\tilde{\lambda}_m(\tilde{\tau}_1))$.
   - If $V > h_{\tilde{\tau}_1}(\tilde{\tau}_1^*)/\tilde{\lambda}_m(0)$ then $\tilde{\tau}_1^*$ is not an event time. In this case, continue performing $\tilde{\tau}_1^* = \tilde{\tau}_1^* + Exp(\tilde{\lambda}_m(0))$ as long as $V > h_{\tilde{\tau}_1}(\tilde{\tau}_1^*)/\tilde{\lambda}_m(0)$, updating $h_{\tilde{\tau}_1}(\tilde{\tau}_1^*)$ each time.

4. Repeat step 3 in decreasing intervals of $(0, \tilde{T} - T - \tilde{\tau}_i^*]$ using $h_{\tilde{\tau}_i}(\tilde{\tau}_i^*)/\tilde{\lambda}_m(\tilde{\tau}_{i-1})$ as the acceptance/rejection criterion.

By simulating the TiDeH $\tilde{N}$ process for sufficiently many replications, we can acquire the mean- and median-forecasts for a retweet cascade from T to $\tilde{T}$ based on the generated event numbers. Intuitively, a prediction interval based on these numbers using some appropriate quantiles can also be obtained to see the range of predicted popularity values.

Before we proceed to presenting the various evaluation metrics used in assessing the performances of different prediction methods proposed in the literature, we note that the SEISMIC in Section 3.3 and the TiDeH model discussed herein have incorporated some basic network information of Twitter into their model formulations and prediction methodologies. Specifically, compared to models of greater complexity, such as the multilevel model of Zaman et al. (2014) which requires the complete network structure to be operable, the SEISMIC and the TiDeH model utilize simplified network information like the observed tweet popularity levels up to a certain time point and the corresponding numbers of followers of tweeters and retweeters to predict the future popularity levels of tweets.

3.5 Performance Evaluation Metrics

In relation to our discussion in later chapters, we shall discuss how the performances of prediction methods under the microscopic level can be assessed, with an additional assumption on that the data consists of uncorrelated retweet time sequences as described in Section 1.3. Furthermore, recall from (2.8.1) that we have denoted the predicted final popularity by $N(\tilde{T})_{\text{pred}}$ and the actual final popularity by $N(\tilde{T})$, for $\tilde{T} = 7$ days.

The performances of different prediction methods can be assessed using evaluation metrics like the root mean squared error (RMSE), the mean absolute error (MAE), the mean absolute percentage error (MAPE), and finally the median absolute percentage error (MdAPE). The formulas to calculate the RMSE and MAE are shown respectively in (3.5.1) and (3.5.2), while the MAPE and MdAPE can be obtained based on the mean and median of (3.5.3).

$$\text{RMSE} = \sqrt{\frac{1}{n_c}\sum_{j=1}^{n_c}\left(N(\tilde{T})_{\text{pred}_j} - N(\tilde{T})_j\right)^2} \tag{3.5.1}$$

$$\text{MAE} = \frac{1}{n_c}\sum_{j=1}^{n_c}\left|N(\tilde{T})_{\text{pred}_j} - N(\tilde{T})_j\right| \tag{3.5.2}$$

$$\text{APE} = \left| \frac{N(\tilde{T})_{\text{pred}_j} - N(\tilde{T})_j}{N(\tilde{T})_j} \right| \tag{3.5.3}$$

We note from (3.5.1), (3.5.2), and (3.5.3) that the subscript j denotes the j^{th} individual cascade, and n_c denotes the total number of retweet cascades under evaluation. This implies that $n_c = 94254$ if the whole test data in Section 1.3 were to be used.

Gneiting (2011) asserted that different prediction functionals are optimal based on different error metrics, and concluded that the RMSE is optimal relative to mean-based prediction whilst the MAE is optimal relative to median-based prediction. We note here that the RMSE will penalize larger errors more, and tends to get heavily distorted by the presence of outliers. This may be problematic for models such as the SEISMIC of Zhao et al. (2015). Under a propounded supercritical regime, the SEISMIC assumes that the process will generate an infinite number of events, which then impedes the evaluation based on the RMSE. In contrast, when one knows the range of actual popularity values, measures based on the absolute errors, such as the MAE, will be relatively more useful.

The APE, or the absolute relative error expressed in term of percentage, is arguably a more informative metric to measure how much the predicted popularity deviates from the actual popularity value, and is particularly useful in comparing the efficiency of prediction methods when the popularity values are distributed very differently. In fact, both Zhao et al. (2015) and Kobayashi and Lambiotte (2016) used the MdAPE to evaluate the prediction performances of their proposed methodologies, since the median is known to be a robust estimator which is more resistant to the presence of outlying APE values. There is, however, a clear pitfall when using such median-based evaluation metric in assessing the performance of a prediction method, since it allows up to half of the predicted popularity values to be arbitrarily bad. Therefore, statistical inferences and conclusions as to how a prediction method outperforms the others should be made using both mean- and median-based evaluation metrics, or in this case the MAPE and MdAPE.

Although the predictive mean and the predictive median are not optimal relative to the MAPE or the MdAPE in general, they are typically much easier to obtain than the functionals that are optimal relative to these metrics[3]. In addition, they are often approximately optimal when the predictive distribution is unimodal. Therefore, they have been widely used in popularity prediction even when the MAPE or the MdAPE is used as the performance evaluation metric.

[3]The functionals that are optimal relative to the MAPE and the MdAPE are the order -1 median and the harmonic median respectively; see Appendix A for more details.

Chapter 4

A Marked Self-Exciting Point Process Model[1]

We have discussed that the sequence of random variables τ_i satisfying $\tau_i < \tau_{i+1}$ can be referred to as the event times. These event times may be associated with some random elements n_i called the event marks. Each (τ_i, n_i) is said to be a marked point, and the sequence of such marked points for $i = 0, 1, \ldots$ is said to be a marked point process. The frequent clustering of retweet events on Twitter suggests that a self-exciting point process might be useful in capturing the retweeting dynamics. Furthermore, as such surge of events seems to be highly correlated with the magnitude of event marks, this prompts us to model the activities based on a marked self-exciting point process.

The model we propose herein to capture the retweeting dynamics and predict the future popularity of tweets is termed the Marked Self-Exciting Process with Time-Dependent Excitation Function, or the MaSEPTiDE for short. It is motivated by the SEISMIC and the TiDeH model, and bears some similarities to them. However, the MaSEPTiDE model has some important advantages. First, its intensity process has a linear form similar to that of the original self-exciting process of Hawkes (1971), and therefore the resulting point process is interpretable as a cluster Poisson process, which implies that the MaSEPTiDE process can be simulated using a cascading algorithm similar to that used for the efficient simulation of Hawkes processes. Second, the estimation of the model and the assessment of its goodness-of-fit can be implemented using principled approaches from the point process theory, and prediction based on the model can also be done properly by exploiting its probabilistic properties, without resorting to ad hoc assumptions such as those needed by the SEISMIC. The model is also found to be able to capture the retweeting dynamics and make accurate popularity predictions based on much shorter observation times

[1]Most of the content shown in this chapter has been published in the *Annals of Applied Statistics*; see Chen and Tan (2018).

than those required by the TiDeH model.

We shall give comprehensive elaborations of the MaSEPTiDE model in this chapter, including some of its limitations and potential for future work. The remainder of this chapter is structured as follows. We first present the form of the intensity assumed by the MaSEPTiDE model in Section 4.1. This is followed by its parameter estimation in Section 4.2, and the assessment of its goodness-of-fit in Section 4.3. The procedures involved to predict the future popularity of tweets are demonstrated in Section 4.4, including a solve-the-equation approach and a simulation-based approach. By applying the proposed methodologies to the Twitter data set, we show the main results, in particular the evaluation of performances among different models, in Section 4.5. Further discussion of the MaSEPTiDE model can be found in Section 4.6, and the concluding remarks are given in Section 4.7.

4.1 Model Formulation

Let $(\tau_i, n_i), i = 1, 2 \ldots$ be a marked point process where $\tau_1 < \tau_2 < \ldots$ denote the event times and $n_1, n_2, \ldots$ denote the respective event marks, originating from (τ^0, n^0). Recall that $\tau^0 = 0$ denotes the posting time of the original tweet, and n^0 denotes the number of followers of the original tweeter. Correspondingly, the event times and marks for $i = 1, 2, \ldots$ refer respectively to the retweet times and the numbers of followers of the retweeters.

Let $N(t) = \sum_{i=1}^{\infty} \mathbb{1}\left\{\tau_i \leqslant t\right\}, t \geqslant 0$ be the associated counting process of retweets, and $\mathcal{F} = \{\mathcal{F}_t; t \geqslant 0\}$, with $\mathcal{F}_t = \sigma\left\{N(t), n^0, (\tau_j, n_j), j = 1, 2, \ldots, N(t)\right\}$, be the natural filtration of the marked point process. In an informal but intuitive notation, the intensity can be written as

$$\lambda(t) = \frac{\mathbb{E}\left[\, \mathrm{d}N(t) \middle| \mathcal{F}_{t-}\right]}{\mathrm{d}t},$$

from which we note that the intensity at any time point is the expected number of events per unit time given the history of the process prior to that time point.

As the evolution of a point process over time is fully determined by its intensity process, a commonly used approach to specify a point process model is to specify the form of the dependence of its intensity process on the prior-t history of the process $\mathcal{F}_{t-}$. Specifically, the intensity assumed by the SEISMIC of Zhao et al. (2015) as presented in (3.3.1) consists of two main component functions, namely the infectivity function $p(\cdot)$ and the memory kernel function $\phi(\cdot)$, both of which are positive functions. Assisted by these two component functions, the function (3.3.1) describes the retweet intensity of a tweet, or the expected number of retweets per unit time, as a product of the infectivity of the original tweet and the accumulated

excitation effects of all previous retweets. Zhao et al. (2015) proposed to estimate the infectivity function $p(\cdot)$ nonparametrically using a kernel smoothing estimator with a triangular kernel. To estimate the memory kernel, they assumed that it is of a power-law decaying form, and that 15 selected retweet cascades follow inhomogeneous Poisson processes with their intensity functions being proportional to the memory kernel. They then estimated the parameters using histogram and complementary cumulative distribution function plots of the retweet times in those 15 cascades.

On the other hand, the TiDeH model of Kobayashi and Lambiotte (2016) assumes an intensity process of the same form as in (3.3.1), except with the further assumption that the infectivity function $p(\cdot)$ is also parametric, and takes a dampened circadian oscillation form. To estimate the infectivity function $p(\cdot)$, Kobayashi and Lambiotte (2016) proposed a two-step approach where a preliminary estimate $p_0(\cdot)$ was first obtained using a kernel method, and then the parametric form of $p(\cdot)$ was fitted to the preliminary estimate by a least squares method.

4.1.1 Intensity Specification

The point process model we propose herein for the purpose of retweeting dynamics modelling has the following intensity function,

$$\lambda(t) = \nu(t) + \sum_{i=1}^{N(t-)} \omega(\tau_i, n_i, t - \tau_i), \tag{4.1.1}$$

where $\nu(\cdot)$ is the *baseline intensity function*, with $\nu(t)$ denoting the part of the event intensity at time t that is due to the initial event at time zero. The function $\omega(\cdot, \cdot, \cdot)$ is the *excitation function*, with $\omega(\tau, n, t - \tau)$ denoting the impact of an event at time τ with mark n on the event intensity at time t, where t is the time since the publication of the original tweet.

Furthermore, both the baseline intensity and the excitation functions are time-dependent and take multiplicatively separable forms as follows,

$$\begin{aligned} &\nu(t) = \alpha\phi(t), \\ &\omega(\tau, n, t - \tau) = p(\tau)r(n)\phi(t - \tau). \end{aligned} \tag{4.1.2}$$

Here $\alpha > 0$ is a constant giving the direct excitation effect of the original tweet, that is, how many retweets it is expected to generate directly. The function $\phi(\cdot)$ is called the *memory kernel function*, which describes how the excitation effect due to the original tweet or a retweet is distributed over time. Similar to Zhao et al. (2015), we require $\phi(\cdot)$ to be a probability density function, so that $\phi(\cdot) \geqslant 0$ and $\int_0^\infty \phi(s)\,\mathrm{d}s = 1$. The function $p(\cdot)$ indicates how the infectivity of a retweet varies over time and is

also called the *infectivity function*, although its influence on the intensity process is different than that of the infectivity function $p(\cdot)$ in (3.3.1). For identifiability, we assume that $p(0) = 1$. The function $r(\cdot)$ is called the *impact function*, and describes the total excitation effect of a retweet attributed to the number of followers of the retweeter. Note, we do not require $\alpha = r(n^0)$, to allow for the potentially different influences of the original tweet and of the retweets.

More specifically, the functions in (4.1.2) are assumed to take the following parametric forms,

$$\begin{aligned} p(\tau;\beta) &= e^{-\beta\tau}, \\ r(n;\gamma) &= \gamma \log(n+1), \\ \phi(t;\delta) &= \frac{\delta_2(\delta_1 - 1)}{\delta_1}\left(1 + \frac{\delta_2 t}{\delta_1}\right)^{-\delta_1}, \end{aligned} \tag{4.1.3}$$

for parameters $\beta \geqslant 0, \gamma \geqslant 0, \delta_1 > 1$, and $\delta_2 > 0$. Here, we have adopted an exponential decay form for the infectivity function, based on the intuition that the infectivity, or the newsworthiness of a retweet, should decay very quickly over time. We further assume that the impact function is linear in the number of followers on a log scale, rather than on the original scale as in Zhao et al. (2015), because of the high degree of right skewness for the distribution of the number of followers (Cha et al., 2010; Kwak et al., 2010; Bakshy et al., 2011). Our choice of the power-law decay form for the memory kernel is motivated by Zhao et al. (2015) and the empirical findings of the heavy-tailed distributions for the human response time in social networks, reported in the literature (Barabasi, 2005; Crane and Sornette, 2008; Zaman et al., 2014).

Similar to Zhao et al. (2015) and Kobayashi and Lambiotte (2016), we also assume that the event marks n_i are i.i.d with a common density function $f(\cdot)$ relative to a suitable reference measure on the space $\mathcal{N}$ of event marks, and moreover, n_i is independent of τ_i and $\mathcal{F}_{\tau_i-}$ for all i. As the excitation function associated with an event is allowed to depend on the time of that event, the model shall be referred to as the *Marked Self-Exciting Process with Time-Dependent Excitation Function*, or the MaSEPTiDE model for short.

At this point we emphasize an important difference between the MaSEPTiDE model we propose and the SEISMIC of Zhao et al. (2015). From (4.1.1), we note that, unlike the SEISMIC, the MaSEPTiDE has an intensity process that is of a linear form similar to the self-exciting process of Hawkes (1971), whose intensity process takes the form as shown in (2.4.1). In fact, if we choose $p(\tau) \equiv 1$ and $r(n) \equiv r$ for a constant r in (4.1.2), then (4.1.1) reduces to the time-varying version of the Hawkes process considered by Chen and Hall (2013, 2016).

4.1.2 Interpretation as a Poisson Cluster Process

The linear structure of the intensity process implies that the MaSEPTiDE can also be interpreted as a Poisson cluster process, as the original Hawkes process or the generalized version with a time-varying background intensity, much of which has been discussed in Section 2.4.1.2.

By this interpretation, immigrants arrive according to a marked inhomogeneous Poisson process with its intensity function equal to the baseline intensity function $\nu(\cdot)$, and event marks distributed according to the density $f(\cdot)$. Once an immigrant with mark n arrives at τ, it starts to independently produce children according to a marked inhomogeneous Poisson process with intensity function $\omega(\tau, n, \cdot) = p(\tau)r(n)\phi(\cdot)$ and event marks distributed according to $f(\cdot)$, so that the total number of children is Poisson distributed with mean $\int_0^\infty \omega(\tau, n, s)\,\mathrm{d}s = p(\tau)r(n)$. Given the total number of children, the waiting times to births of the children are i.i.d with a common density function $\phi(\cdot)$, if the order of births is ignored.

Moreover, once an offspring of any generation is born, say at time τ' and with mark n', it starts to independently produce children of its own according to a similar marked inhomogeneous Poisson process with intensity function $\omega(\tau', n', \cdot)$ and event marks distributed according to $f(\cdot)$. The events of the MaSEPTiDE process by time t consist of all immigrants and offspring of any generation that have arrived by time t. This Poisson cluster process interpretation implies an efficient recursive cascading algorithm to simulate the MaSEPTiDE process, and has important implications for simulation-based predictions by the process. Figure 4.1.1 shows a graphical representation of the discussed cluster process, where each circular point represents an immigrant or offspring with a certain event time and mark, each line connecting two points represents the parent-offspring relationship, each vertical arrow represents the collective shift from one generation to another, and the vertical dotted line represents the continuation of the process for future generations.

Because of the Poisson cluster interpretation, the memory kernel function $\phi(\cdot)$ in the MaSEPTiDE can also be called the *offspring density function*, and the function $p(\cdot)r(\cdot)$ might be interpreted as the *branching ratio function* which specifies how the branching ratio, that is, the average number of direct offspring from an individual (be it an immigrant or an offspring), depends on the birth time and event mark of the individual. In contrast, the functions $p(\cdot)$ and $\phi(\cdot)$ in the SEISMIC or the TiDeH model do not permit such a neat interpretation.

It might also be of interest to note the difference between the treatments of the background intensity in the MaSEPTiDE model and in the Hawkes process model with a time-varying background intensity. In the former model, we require the baseline intensity function to be proportional to the memory kernel $\phi(\cdot)$, while

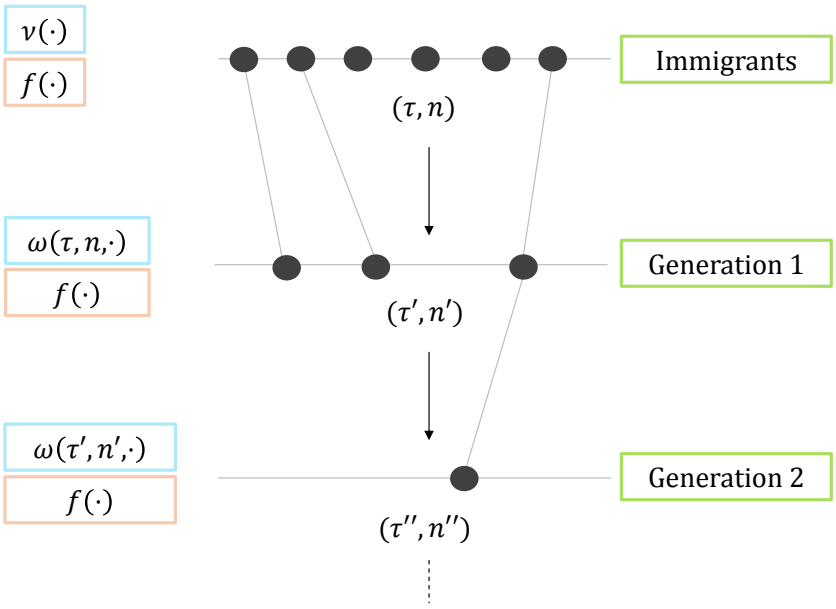

Figure 4.1.1: A cluster process representation of the MaSEPTiDE process. Each immigrant arrives according to the baseline intensity $\nu(\cdot)$ and mark density $f(\cdot)$, and each offspring is generated according to the excitation function $\omega(\cdot, \cdot, \cdot)$ and mark density $f(\cdot)$, from its parent of the preceding generation. Events beyond generation 2 are represented by the vertical dotted line.

in the latter, the background intensity and the memory kernel can take different shapes. The advantage of our treatment is that it leads to a more parsimonious model, while the time-varying background intensity model can easily accommodate nonstationarity, such as that due to the diurnal patterns of human activity levels.

4.2 Parameter Estimation

Before we can use the MaSEPTiDE model to predict the future number of events, we need to first estimate the model parameters. Since the event marks are assumed to be i.i.d, their distribution can simply be estimated by the empirical distribution of n_i, for $i = 1, 2, \ldots, N(T)$. The main estimation problem is to estimate the parameter vector $\theta = (\alpha, \beta, \gamma, \delta_1, \delta_2)^\top$. To this end, we shall use the ML approach, which has been previously discussed in Section 2.5.

By the point process theory, the likelihood of the MaSEPTiDE process based on observations over the interval $[0, T]$, where T denotes the censoring time, takes the following form

$$L(\theta) = \left\{ \prod_{i=1}^{N(T)} \lambda(\tau_i) \right\} \exp\left(- \int_0^T \lambda(t)\, \mathrm{d}t \right) \prod_{i=1}^{N(T)} f(n_i), \tag{4.2.1}$$

where $\lambda(\cdot)$ depends on the parameters through (4.1.1)-(4.1.3), and $f(\cdot)$ denotes the

event mark density, which is assumed to be free of the parameters θ. The likelihood of the process without marks has also been shown in (2.5.1).

To compute the ML estimator of the parameter vector θ using general-purpose numerical optimization routines, the efficient evaluation of the likelihood function or its logarithm is very important. For this purpose, we need to be able to evaluate the definite integral of the intensity function in (4.2.1) efficiently. Due to the linear structure of the intensity function, the integral of the intensity function can be shown to take an explicit form similar to the intensity function itself, and therefore can be exactly computed without resorting to numerical quadrature routines. To show this, it is convenient to use the random measure interpretation of a marked point process. That is, we interpret

$$N(\mathrm{d}\tau, \mathrm{d}n) = \sum_{i=1}^{\infty} \delta_{(\tau_i, n_i)}(\mathrm{d}\tau, \mathrm{d}n)$$

as a random measure on $[0, \infty) \times \mathcal{N}$, so that the intensity in (4.1.1) can be written as

$$\begin{aligned}\lambda(t) &= \nu(t) + \sum_{i=1}^{N(t-)} \omega(\tau_i, n_i, t - \tau_i) \\ &= \nu(t) + \int_{(0,t)\times\mathcal{N}} \omega(\tau, n, t - \tau) N(\mathrm{d}\tau, \mathrm{d}n).\end{aligned}$$

Therefore, by *Fubini's theorem*, a change of variables, and the assumed forms of the functions ν, ω and ϕ, we have

$$\begin{aligned}\int_0^T \lambda(t)\,\mathrm{d}t &= \int_0^T \nu(t)\,\mathrm{d}t + \int_0^T \int_{(0,t)\times\mathcal{N}} \omega(s, n, t - s) N(\mathrm{d}s, \mathrm{d}n)\,\mathrm{d}t \\ &= \int_0^T \nu(t)\,\mathrm{d}t + \int_{(0,T)\times\mathcal{N}} \int_s^T \omega(s, n, t - s)\,\mathrm{d}t N(\mathrm{d}s, \mathrm{d}n) \\ &= \int_0^T \nu(t)\,\mathrm{d}t + \int_{(0,T)\times\mathcal{N}} \int_0^{T-s} \omega(s, n, t)\,\mathrm{d}t N(\mathrm{d}s, \mathrm{d}n) \\ &= \alpha\Phi(T) + \sum_{i=1}^{N(T-)} p(\tau_i) r(n_i) \Phi(T - \tau_i),\end{aligned}$$

where the integrated memory kernel function $\Phi(\cdot)$ can be written as

$$\Phi(t) = \Phi(t; \delta) = \int_0^t \phi(s; \delta)\,\mathrm{d}s = 1 - \left(1 + \frac{\delta_2 t}{\delta_1}\right)^{-\delta_1 + 1}, \quad t \geqslant 0. \tag{4.2.2}$$

From the separable form of the likelihood function in (4.2.1) and the assumption that the event mark distribution does not depend on the parameter vector

θ, the ML estimation of the tweet specific parameters θ can be based on maximizing the logarithm of the part of the likelihood that does not involve $f(\cdot)$, as shown previously in (2.5.2). In practice, the maximization can be done numerically using various general-purpose optimization routines. As previously mentioned in Section 2.5, Newton methods such as the `BFGS` method can be used in achieving this purpose. We have, however, used the more robust downhill simplex method of Nelder and Mead (1965) in our numerical experiments, which is the default method used by the function `optim` in the R software environment for statistical computing (R Core Team, 2016).

4.3 Goodness-of-Fit Assessment

The assessment of the goodness-of-fit of models to historical data can guide us to seek models that can describe the observed data well and therefore serves as the basis of predictions for future observations. To assess the goodness-of-fit of the MaSEPTiDE model, we shall use the residual point process approach based on Papangelou's random time change theorem, detailed in Section 2.6.

By the time change theorem, with $\Lambda(t) = \int_0^t \lambda(s)\,\mathrm{d}s$ denoting the cumulative intensity process, the transformed process $N(\Lambda^{-1}(t))$ is a Poisson process with unit rate or equivalently, the random times $\Lambda(\tau_i)$, $i = 1, 2, \ldots$, will be the event times of a unit rate Poisson process. Therefore, if the MaSEPTiDE with the parameters θ set to their ML estimates $\hat{\theta}$ is a sufficient model for the observed event times up to the censoring time T, then the transformed event times, $\hat{\Lambda}(\tau_i)$, $i = 1, 2, \ldots, N(T)$ should be approximately equal in distribution to the event times of a unit rate Poisson process up to time $\hat{\Lambda}(T)$. Here, $\hat{\Lambda}(t)$, $t > 0$ is the plugin estimate of the cumulative intensity $\Lambda(t;\theta) = \int_0^t \lambda(s;\theta)\,\mathrm{d}s$, that is,

$$\hat{\Lambda}(t) = \Lambda(t;\hat{\theta}) = \hat{\alpha}\Phi(t;\hat{\delta}) + \sum_{i=1}^{N(t-)} p(\tau_i;\hat{\beta}) r(n_i;\hat{\gamma}) \Phi(t-\tau_i;\hat{\delta}),$$

with $p(\cdot)$ and $r(\cdot)$ defined in (4.1.3), and $\Phi(\cdot)$ defined as in (4.2.2).

We have highlighted in Section 2.6 that the conditional distribution of the event times of a Poisson process in a fixed interval, given the total number of events in the interval, is equal in distribution to the order statistics of the same number of i.i.d random variables uniformly distributed in the interval. Thus, to assess the goodness-of-fit of the MaSEPTiDE model, we can assess the uniformity of the transformed event times $\hat{\Lambda}(\tau_i)$, $i = 1, 2, \ldots, N(T)$, in the interval $(0, \hat{\Lambda}(T)]$ using tests like the Kolmogorov-Smirnov test, or informally using graphical approaches such as the histogram or the quantile-quantile plots. A similar analysis was performed by Ogata (1988) to assess the goodness-of-fit of point process models on earthquake data.

4.4 Predicting the Popularity

Given observations up to T, to predict the number of events from T to a future time point $\tilde{T} > T$, one commonly uses its conditional expectation or its conditional median. To obtain the conditional expectation, we can use either a solve-the-equation approach or a simulation-based approach. The former approach involves deriving a functional equation satisfied by the conditional expectation as a function of a future time point, solving the equation, and evaluating the solution function at the desired time point. The latter approach involves simulating the sample path of the MaSEPTiDE on the time interval $(T, \tilde{T}]$ conditional on the observations up to time T for a large number of times, counting the number of events on each simulated sample path, and using the average of the simulated event counts to approximate its expectation. While the first approach is computationally less expensive, the solution of the functional equation is not always easy to obtain. For the second approach, although it is relatively less efficient, especially if the process to be simulated has a large expected number of events, it is more robust than the first approach. To obtain the conditional median, the only option seems to be a simulation-based approach, which involves simulating the conditional sample path of the MaSEPTiDE process a large number of times and extracting the median of the resultant empirical distribution of the number of events in the time interval $(T, \tilde{T}]$.

4.4.1 Translated Intensity

Both solve-the-equation approach and simulation-based approach rely on the observation that, conditional on the history of the MaSEPTiDE process up to time T, its future evolution is the same as that of another MaSEPTiDE process with a different baseline intensity function and a similar excitation function. Specifically, if we let $\tilde{N}(t) = N(T+t) - N(T)$, $t \geqslant 0$, and $\tilde{\tau}_j = \tau_{N(T)+j} - T$, $\tilde{n}_j = n_{N(T)+j}$ for $j = 1, 2, \ldots,$ where $\tilde{\mathcal{F}}_t = \mathcal{F}_{T+t}$, $t \geqslant 0$, then the $\tilde{\mathcal{F}}$-intensity process of $\tilde{N}(t)$ is given by

$$\begin{aligned}\tilde{\lambda}(t) = \lambda(T+t) &= \nu(T+t) + \sum_{j=1}^{N(T)} \omega(\tau_j, n_j, T+t-\tau_j) + \sum_{j=N(T)+1}^{N(T+t-)} \omega(\tau_j, n_j, T+t-\tau_j) \\ &= \tilde{\nu}(t) + \sum_{j=1}^{\tilde{N}(t-)} \tilde{\omega}(\tilde{\tau}_j, \tilde{n}_j, t-\tilde{\tau}_j),\end{aligned}$$

where $\tilde{\nu}(\cdot)$ denotes the function

$$\tilde{\nu}(t) = \nu(T+t) + \sum_{j=1}^{N(T)} \omega(\tau_j, n_j, T+t-\tau_j), \tag{4.4.1}$$

and $\tilde{\omega}(\cdot,\cdot,\cdot)$ denotes the function

$$\tilde{\omega}(\tau, n, t) = \omega(T+\tau, n, t) = p(T+\tau)r(n)\phi(t) \equiv \tilde{p}(\tau)r(n)\phi(t). \tag{4.4.2}$$

Therefore, $\tilde{N}(t)$, $t \geqslant 0$ is a MaSEPTiDE process with baseline intensity function $\tilde{\nu}$ and excitation function $\tilde{\omega}$ given in (4.4.1) and (4.4.2) respectively. The excitation function $\tilde{\omega}$ has a similar separable form as ω, with r and ϕ the same as before, and the function $\tilde{p}$ equals to a time shift of the previous infectivity function, that is, $\tilde{p}(\tau) = p(T+\tau)$.

4.4.2 Solve-the-Equation Approach

To calculate the expected number of events $\mathbb{E}[N(\tilde{T}) - N(T)|\mathcal{F}_T]$ without resorting to simulations, we first note from the definition of the conditional intensity that,

$$\begin{aligned}
&\mathbb{E}\left[N(\tilde{T}) - N(T)\middle|\mathcal{F}_T\right] = \mathbb{E}\left[\tilde{N}(\tilde{T}-T)\middle|\mathcal{F}_T\right] \\
&= \mathbb{E}\left[\int_0^{\tilde{T}-T} \tilde{\lambda}(s)\,\mathrm{d}s\middle|\mathcal{F}_T\right] = \int_0^{\tilde{T}-T} \mathbb{E}\left[\tilde{\lambda}(s)\middle|\mathcal{F}_T\right]\,\mathrm{d}s \\
&= \int_0^{\tilde{T}-T} \bar{\lambda}(s)\,\mathrm{d}s,
\end{aligned} \tag{4.4.3}$$

with $\bar{\lambda}(s) = \mathbb{E}[\tilde{\lambda}(s)|\mathcal{F}_T]$ denoting the mean intensity function of $\tilde{N}(t)$ given $\mathcal{F}_T$. By the independence between event marks and previous event times, we have

$$\begin{aligned}
\bar{\lambda}(t) &= \mathbb{E}\left[\tilde{\lambda}(t)\middle|\mathcal{F}_T\right] \\
&= \mathbb{E}\left[\tilde{\nu}(t) + \int_{(0,t)\times\mathcal{N}} \tilde{\omega}(\tau, n, t-\tau)\tilde{N}(\,\mathrm{d}\tau,\,\mathrm{d}n)\middle|\mathcal{F}_T\right] \\
&= \mathbb{E}\left[\tilde{\nu}(t) + \int_{(0,t)\times\mathcal{N}} \tilde{p}(\tau)r(n)\phi(t-\tau)\tilde{\lambda}(\tau)\,\mathrm{d}\tau\,\mathrm{d}F(n)\middle|\mathcal{F}_T\right] \\
&= \tilde{\nu}(t) + \int_{\mathcal{N}} r(n)\,\mathrm{d}F(n) \int_0^t \tilde{p}(\tau)\phi(t-\tau)\mathbb{E}\left[\tilde{\lambda}(\tau)\middle|\mathcal{F}_T\right]\,\mathrm{d}\tau \\
&= \tilde{\nu}(t) + R\int_0^t \tilde{p}(\tau)\phi(t-\tau)\bar{\lambda}(\tau)\,\mathrm{d}\tau,
\end{aligned} \tag{4.4.4}$$

where we have also used $\tilde{N}(\,\mathrm{d}\tau,\,\mathrm{d}n)$ to denote the associated random measure again, and F denotes the distribution of the i.i.d event marks, while

$$R = \mathbb{E}\left[r(n_i)\right] = \int_{\mathcal{N}} r(n)\,\mathrm{d}F(n) \tag{4.4.5}$$

is the expected total excitation effect due to an event, or the expected response. Note how the expected response here differs from that of the SEISMIC and the

TiDeH model with $R = \mathbb{E}[n_i]$. Despite the difference, our expected response here still bears a closer resemblance to that of the TiDeH model as both are influenced by the previous instances of the numbers of followers, unlike that of the SEISMIC which depends solely on the average number of followers in the training data set.

In general, we need to solve the integral equation in (4.4.4) numerically to obtain $\bar{\lambda}(t)$ on $[0, \tilde{T} - T]$ and use it in finding the conditional expectation of the number of events in (4.4.3). One method to solve (4.4.4) is to approximate $\bar{\lambda}(t)$ by a flexible parametric function and identify the parameters by requiring both sides of the equation to be equal or approximately equal at sufficiently many points in the interval $[0, \tilde{T} - T]$. Examples of the flexible parametric functions to approximate $\bar{\lambda}(t)$ include a B-spline function with a specified order and knot sequence, or a truncated Fourier series. In both cases, the unknown parameters of the approximating function can be obtained by solving a linear equation of the unknown parameters. In practice, we would try approximating functions with increasing flexibility until convergence in the solution is achieved.

We have selected the B-spline function as a method to find $\bar{\lambda}(t)$ for its ease of implementation and computational stability. This method has been shown in (3.4.5) for the TiDeH model, although the implementation varies slightly. Similar to the TiDeH model, the set of B-spline basis functions on the interval $(0, \tilde{T} - T]$ shall be denoted by $B(t) = (B_1(t), B_2(t), \ldots, B_k(t))^\top$, and we assume that $\bar{\lambda}(t) \approx B(t)^\top \eta$ for a k-vector η. Plugging this into (4.4.4) yields the following equation of η,

$$B(t)^\top \eta = \tilde{\nu}(t) + \left\{ R \int_0^t \tilde{p}(\tau) \phi(t - \tau) B(\tau)^\top \eta \, \mathrm{d}\tau \right\}. \tag{4.4.6}$$

To solve (4.4.6) for η, we need to evaluate both sides of (4.4.6) at sufficiently many ($\geqslant k$) t values over the interval $(0, \tilde{T} - T]$, and solve the resulting overdetermined linear system using the method of least squares. Once η is obtained, the predicted value can be calculated from

$$\left\{ N(\tilde{T}) - N(T) \right\}_{\text{pred}} = \left(\int_0^{\tilde{T}-T} B(t) \, \mathrm{d}t \right)^\top \eta.$$

In evaluating the integrals in (4.4.6), we often need to use numerical quadrature routines. In our case, we have used the R function `integrate` for this purpose.

4.4.3 Simulation-Based Approach

To simulate the MaSEPTiDE $\tilde{N}$ process over the interval $(0, \tilde{T} - T]$, we can use the following *cascading algorithm*, which is a generalization of that used for the simulation of nonstationary self-exciting point processes (Chen and Hall, 2013, 2016). A

similar algorithm has also been used recently by Chen and Stindl (2018) to simulate renewal Hawkes processes.

1. Simulate an inhomogeneous Poisson process N^0 with time-varying intensity $\tilde{\nu}(t)$ on $(0, \tilde{T}-T]$ and denote the event times by $\tau_j^{(0)}$, $j = 1, 2, \ldots, N^0(\tilde{T}-T)$.

2. Generate the associated event marks $n_j^{(0)}$ independently from the event mark distribution F and call the events $(\tau_j^{(0)}, n_j^{(0)}), j = 1, 2, \ldots, N^0(\tilde{T}-T)$ generation 0 events.

3. For each generation 0 event $(\tau_j^{(0)}, n_j^{(0)})$, simulate an inhomogeneous marked Poisson process N_j^1, with intensity function $\tilde{\omega}(\tau_j^{(0)}, n_j^{(0)}, \cdot)$ and event mark distribution F, on the interval $(0, \tilde{T} - T - \tau_j^{(0)}]$ and denote the corresponding events by $(\tau_{jk}^{(1)}, n_{jk}^{(1)}), k = 1, 2, \ldots, N_j^1(\tilde{T} - T - \tau_j^{(0)})$. We refer the collection of events $\{(\tau_j^{(0)} + \tau_{jk}^{(1)}, n_{jk}^{(1)}); k = 1, 2, \ldots, N_j^1(\tilde{T}-T-\tau_j^{(0)}), j = 1, 2, \ldots, N^0(\tilde{T}-T)\}$ to as generation 1 events.

4. Continue generating events of generations $2, 3, \ldots$ similarly on intervals of decreasing lengths, until a generation has no events.

5. The events of all generations are pooled together to form the collection of all events of the MaSEPTiDE $\tilde{N}$ process on the interval $(0, \tilde{T} - T]$.

The algorithm shown above requires the simulation of inhomogeneous Poisson processes, which can be achieved using the thinning algorithm of Lewis and Shedler (1979). For that purpose, we have used the simulation procedures described in Section 2.7, which make use of the `simPois` function in the `IHSEP` package. Furthermore, our implementation of the above cascading algorithm is also based on a simple modification of a function in the same `IHSEP` package, namely the `simHawkes1` function.

It should be clear from the cascading algorithm shown above that the inhomogeneous Poisson process N^i essentially generates $(\tau^{(i)}, n^{(i)})$ where $i = 0, 1, 2, \ldots$ denotes the generation number, with the respective subscripts $j, jk, jkl, \ldots$. To help visualizing the implementation of the algorithm, we demonstrate in Figure 4.4.1 how the events beyond the censoring time T, up to the prediction time point $\tilde{T}$, can be generated. Similar to Figure 4.1.1, each circular point represents an event with a certain time and mark, each line connecting two points represents the parent-offspring relationship, each vertical arrow represents the collective shift from one generation to another, and the vertical dotted line represents the continuation of the process for future generations.

To predict the number of events in the interval $(T, \tilde{T}]$, we simulate the sample path of the process $\tilde{N}(t)$ over the interval $(0, \tilde{T} - T]$ for a large number of times,

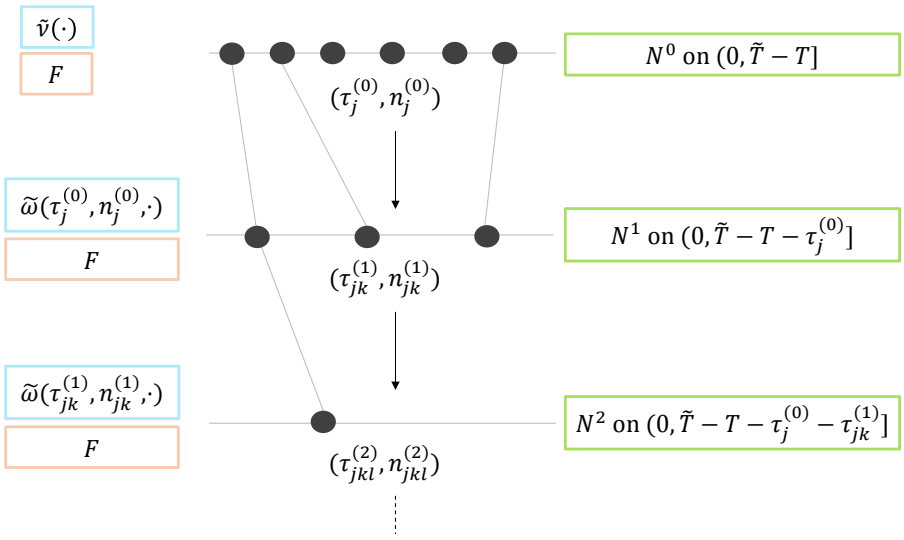

Figure 4.4.1: The cascading algorithm used to simulate the MaSEPTiDE $\tilde{N}$ process, where the events of i^{th} generation are simulated from N^i, and are denoted by $(\tau^{(i)}, n^{(i)})$ for $i = 0, 1, 2, \ldots$ with subscripts $j, jk, jkl, \ldots$. Each generation 0 event arrives according to the baseline intensity $\tilde{\nu}(\cdot)$ with mark sampled from the distribution F, and each event beyond generation 1 is generated according to the excitation function $\tilde{\omega}(\cdot, \cdot, \cdot)$ with mark similarly sampled from the distribution F. Events beyond generation 2 are represented by the vertical dotted line.

say 100, and count the number of events on each simulated sample path. The mean or median of these simulated event numbers will then be our point prediction of the number of events of the MaSEPTiDE process in the interval $(T, \tilde{T}]$. It is also worth noting that with sufficiently many replications, the mean of the simulated event numbers should be consistent with the prediction based on that of the solve-the-equation approach.

In practice, when we use the fitted model to make predictions, whether by using the solve-the-equation approach or by using the simulation-based approach, the unknown functions $\tilde{\nu}$ and $\tilde{\omega}$, and the event mark distribution F need to be replaced by their respective estimators. In our numerical experiments, we have used the plugin estimators $\tilde{\nu}(\cdot; \hat{\theta})$ and $\tilde{\omega}(\cdot, \cdot, \cdot; \hat{\theta})$ for $\tilde{\nu}$ and $\tilde{\omega}$, and the empirical distribution function $\hat{F}$ of the event marks $n_1, n_2, \ldots, n_{N(T)}$ for F. One implication is that the constant in (4.4.5) is set to

$$\hat{R} = \int_{\mathcal{N}} r(n) \, \mathrm{d}\hat{F}(n) = \frac{1}{N(T)} \sum_{i=1}^{N(T)} r(n_i).$$

Finally, we note that, if the target of prediction is the total number of events of the process N in the interval $(0, \tilde{T}]$, then we simply add the observed number of events in $(0, T]$, that is, $N(T)$, to the predicted number of events in $(T, \tilde{T}]$, as in (2.8.1).

4.5 Application to the Tweet Data

Here, we report the results of applying the proposed model and inference methodologies presented in Section 4.1-4.4 to the Twitter data described in Section 1.3. The performance of our prediction methods is also compared to those of the SEISMIC and the TiDeH model. The implementation details for the SEISMIC and the TiDeH model have been given in Section 3.3 and Section 3.4 respectively.

4.5.1 Typical Parameter Values

We fitted the MaSEPTiDE model to all the retweet cascades in the training data set with different censoring times, using the ML method described in Section 4.2. The estimated parameter values with the censoring time of seven days are found to be highly skewed, with the median estimates of α, β, γ, δ_1 and δ_2 equal to 48.349, 0.072, 7.209, 1.416, and 0.007 respectively. To have some idea about the typical parameter values found in practice, we consider five random cascades from the training data set, depicted previously in Figure 1.3.2, and display their estimated parameter values in Table 4.5.1. Their final popularity values have also been included for statistical inference.

Table 4.5.1: Fitted parameter values and the actual final popularity for each of the sample cascades shown in Figure 1.3.2. The parameter values are useful in gaining insights on the retweet activities.

Sample cascade	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\gamma}$	$\hat{\delta}_1$	$\hat{\delta}_2$	$N(\tilde{T})$
1	5.711	0.024	1.455	1.254	0.173	159
2	3.075	0.021	6.351	1.414	0.029	85
3	58.136	0.246	1.144	1.490	0.001	55
4	8.209	0.031	2.095	1.444	0.040	74
5	4.173	0.019	5.049	1.229	0.046	89

The estimated values of the parameter β suggest very fast decays of infectivity, with the times taken for the infectivity to drop to 1% of the initial levels vary from about 19 seconds ($\log(100)/0.246 = 18.7$ seconds) in sample cascade 3 to about 4 minutes ($\log(100)/0.019 = 242.4$ seconds) in sample cascade 5. While the estimated values of the shape parameter of the memory kernel δ_1 are more or less similar to each other, the scale parameter δ_2 has substantially more variable values. In particular, the extremely small $\hat{\delta}_2$ value of 0.001 in sample cascade 3 implies a very long range memory effect, which, together with a relatively large $\hat{\beta}$ value, suggest that the later retweets are more likely to be generated by the original tweet or retweets within the first few seconds of the original tweet, if any. In contrast, the $\hat{\delta}_2$ value for sample cascade 1 is 0.173, which implies that the later retweets are more likely to be generated by more recent retweets.

The estimated values of the scale parameter α of the baseline intensity, together with the values of the δ parameters and the final popularity, suggest highly variable proportions of generation 0 retweets, from 3.4% ($= 5.711\Phi(\tilde{T}; 1.254, 0.173)/159$) in sample cascade 1 to nearly 100% ($= 58.136\Phi(\tilde{T}; 1.490, 0.001)/55$) in sample cascade 3. On another note, the estimated γ values on the five sample cascades also seem to have quite substantial variation, with the increase in the excitation effect associated with one unit increase in the number of followers of a retweeting account on the log scale varies from 1.144 to 6.351 units. The overall shapes of parameters from the infectivity function $p(\cdot)$, the impact function $r(\cdot)$, and the memory kernel function $\phi(\cdot)$ for these five sample cascades have been appended in Figure B.1.1 for reference.

4.5.2 Model Goodness-of-Fit

By the goodness-of-fit assessment method described in Section 4.3, we tested the uniformity of the point process residuals $\hat{\Lambda}(\tau_i)$ over the interval $(0, \hat{\Lambda}(T)]$ using the Kolmogorov-Smirnov test. At significance levels of 0.01 and 0.05 with different censoring times, the percentages of all the cascades from the training data set where the estimated MaSEPTiDE model passes the residual uniformity test are shown in Table 4.5.2. From this table we note that the percentage of cascades from which the estimated model passes the test decreases when the censoring time increases. This is to be expected as the amount of data increases with the censoring time, implying that the difficulty of finding a fitting model also increases.

Table 4.5.2: The percentages of cascades in the training data set where the MaSEPTiDE model passes the goodness-of-fit test at different significance levels and censoring times. At significance levels of 0.01 and 0.05, the percentages of cascades passing the test using data accumulated in the first 12 hours are considerably high, at 82% and 78% respectively, which indicate a good fit of the model to the data.

Significance level	Censoring time (hours)						
	2	4	6	8	10	12	168
0.01	92.0%	88.2%	85.8%	84.2%	82.8%	81.8%	74.9%
0.05	89.3%	84.7%	81.9%	80.1%	78.5%	77.5%	69.2%

At significance level of 0.01, when fitted to the complete retweet cascade data, that is, with the censoring time of 168 hours or seven days, the MaSEPTiDE model passes the goodness-of-fit test on roughly 75% of the cascades. In contrast, by the censoring time of 12 hours, the MaSEPTiDE model passes the goodness-of-fit test on the majority of the cascades, at roughly 82%. Given that the majority of the retweets, or 80% on average, have already occurred within the first 12 hours since the publications of the original tweets, as shown in Table 1.3.2, we conclude that the MaSEPTiDE model is able to describe the retweeting dynamics reasonably well.

4.5.3 Prediction Performance Comparisons

For all the tweets in the test data set, we applied the fitted MaSEPTiDE model with the retweet cascades censored at different times to predict their final popularity, using the prediction methods discussed in Section 4.4. For the purpose of comparison, we also obtained the predictions based on the SEISMIC of Zhao et al. (2015) and the TiDeH model of Kobayashi and Lambiotte (2016). We only report the results of comparisons with these two methods, because they were found to outperform other methods in the literature, such as those reported in Crane and Sornette (2008), Agarwal et al. (2009), Szabo and Huberman (2010), and Gao et al. (2015), both in our numerical experiments and in the works of Zhao et al. (2015) and Kobayashi and Lambiotte (2016). We further note here that although the hybrid method of prediction proposed by Mishra et al. (2016) performs relatively better than the SEISMIC and the TiDeH model, the method has not been included for comparisons since it requires additional features such as those based on the other retweet cascades in the discriminative step to obtain accurate popularity predictions.

Our point prediction of the final popularity of a tweet, or the total number of retweets by time $\tilde{T} = 7$ days, using the MaSEPTiDE model estimated with the retweet cascade observed up to the censoring time T, is given by $N(\tilde{T})_{\text{pred}} = N(T)+(N(\tilde{T})-N(T))_{\text{pred}}$ as in (2.8.1), where $(N(\tilde{T})-N(T))_{\text{pred}}$ is obtained either as the conditional expectation using the solve-the-equation approach or the simulation-based approach, or as the conditional median using the simulation-based approach.

We have mentioned in Section 4.4.3 regarding the consistency of both the solve-the-equation and simulation-based approaches when using the conditional expectation as a point prediction. Our numerical experiments have confirmed that the two approaches produce identical predictions up to a negligible numerical error, as expected. For the majority of the retweet cascades, a moderately large number of simulation replications, for instance 100 or even 50, was enough to produce a prediction consistent with that by the solve-the-equation approach. The same set of simulation replications was used to calculate the conditional median. Justification on the suggested number of simulation replications can be found in Appendix C.2.

To assess the performance of the conditional mean predictions, we shall first follow the literature (Zhao et al., 2015; Kobayashi and Lambiotte, 2016) and use the absolute percentage error (APE) shown in (3.5.3) to compare the accuracy of predictions by different models. Each prediction method under evaluation was applied to each of the retweet cascades in the test data set with censoring times $T = 2, 4, \ldots, 12$ hours, as most retweets would have already occurred within the first few hours since the publication of the original tweet, as exhibited in Table 1.3.2.

For each censoring time, we calculated the APEs of the conditional mean predic-

tions based on the proposed model and the two competing models. The predictions by the SEISMIC approach were calculated using the R package `seismic`. The predictions by the TiDeH model approach were calculated using the algorithm described in Section 3.4, with the window size parameter in the estimation step set to one hour. Due to the lack of a principled approach to select the window size, we chose this value based on experimenting with several different values and selecting the one that seemed to produce reasonable estimates of the infectivity function by visual inspection.

The boxplots of the APEs of the conditional mean predictions by the three models at different censoring times are shown in Figure 4.5.1. In each boxplot, the

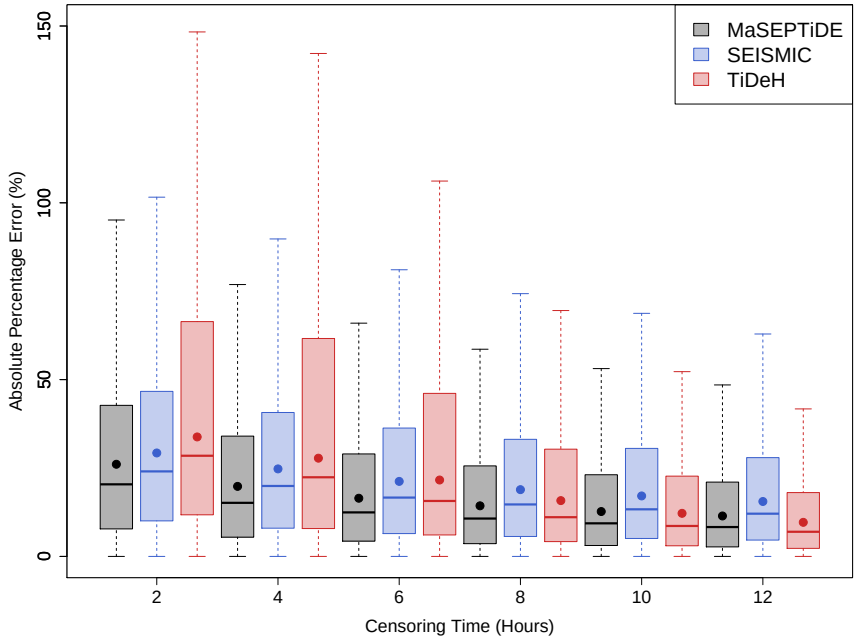

Figure 4.5.1: Boxplots of the APEs of predictions by the MaSEPTiDE model, the SEISMIC and the TiDeH model, at censoring times $T = 2, 4, \ldots, 12$ hours. The horizontal thick bar in each boxplot indicates the median while the circular point indicates the respective mean of APEs. Both the median and mean of APEs demonstrate the superior performance and stability of the MaSEPTiDE model.

horizontal thick bar indicates the median APE (MdAPE), and the circular point indicates the mean APE (MAPE). The actual values of the MdAPEs and MAPEs at all the censoring times considered are also provided in Table 4.5.3. From Figure 4.5.1 and Table 4.5.3, the MaSEPTiDE model seems to have consistently smaller MdAPE and MAPE at each T than the SEISMIC. Compared to the TiDeH model, the MaSEPTiDE model has clearly better performances when $T = 2, 4, 6$ hours, both by the MdAPEs and MAPEs. The performances of these two models are comparable when $T = 8$ hours, but the MaSEPTiDE model appears to slightly underperform the TiDeH model when $T = 10, 12$ hours.

Note how the conditional mean predictions have been used here in obtaining

Table 4.5.3: Median and mean APEs of the popularity predictions by different approaches with observations up to various censoring times T. The MaSEPTiDE model consistently performs better than the SEISMIC at all the censoring times based on both the median and mean of APEs. The MaSEPTiDE model performs better at earlier censoring times $T = 2, 4, 6$ hours compared to the TiDeH model, is comparable when $T = 8$ hours, but underperforms the TiDeH model when $T = 10, 12$ hours.

T	Median APE (%)			Mean APE (%)		
(hours)	MaSEPTiDE	SEISMIC	TiDeH	MaSEPTiDE	SEISMIC	TiDeH
2	19.1	22.8	23.7	26.1	29.3	33.8
4	13.9	18.6	17.1	19.8	24.8	27.8
6	11.2	15.1	12.7	16.5	21.2	21.6
8	9.5	13.1	9.3	14.3	18.9	15.8
10	8.2	11.7	7.4	12.7	17.1	12.2
12	7.3	10.6	5.9	11.4	15.5	9.6

the MdAPEs and MAPEs of the models at the various censoring times, although the optimal functionals relative to the MdAPE and MAPE, as demonstrated in Appendix A, are the harmonic median and the order -1 median respectively. We have not used such prediction functionals primarily due to the computational complexity involved for their acquisitions, and the fact that nearly all past instances of existing works in the literature have consistently used the predictive mean as the point prediction. However, the APE as a prediction error measure is not consistent with the feature of the predictive distribution used as a point prediction here, which is the expectation. A more appropriate error measure when the conditional expectation is used as the point prediction, as conferred in Section 3.5, is the squared error. Therefore, we also calculated the prediction squared errors by the three models. The boxplots of the squared errors by the three models at different censoring times are shown in the left panel of Figure 4.5.2.

As the models can occasionally produce extremely large predictions, even infinity in the case of SEISMIC when cascades falling under the supercritical regime are prevalent, the outlying values have not been shown in the boxplots of Figure 4.5.1 and Figure 4.5.2 for better visualization. The mean squared prediction errors (MSEs) at different censoring times are indicated by the circular points in the boxplots, and their squared roots, that is, the root mean squared errors (RMSEs), are shown in Table 4.5.4. From Figure 4.5.2 and Table 4.5.4, we note that, using the RMSE as the performance measure, the MaSEPTiDE model outperforms the SEISMIC at all the censoring times, and similar to the conclusion drawn based on the median of APEs, the MaSEPTiDE model again, outperforms the TiDeH model when $T = 2, 4, 6$ hours but slightly underperforms when $T = 8, 10, 12$ hours. In comparison, the SEISMIC only outperforms the TiDeH model at $T = 2, 4$ hours.

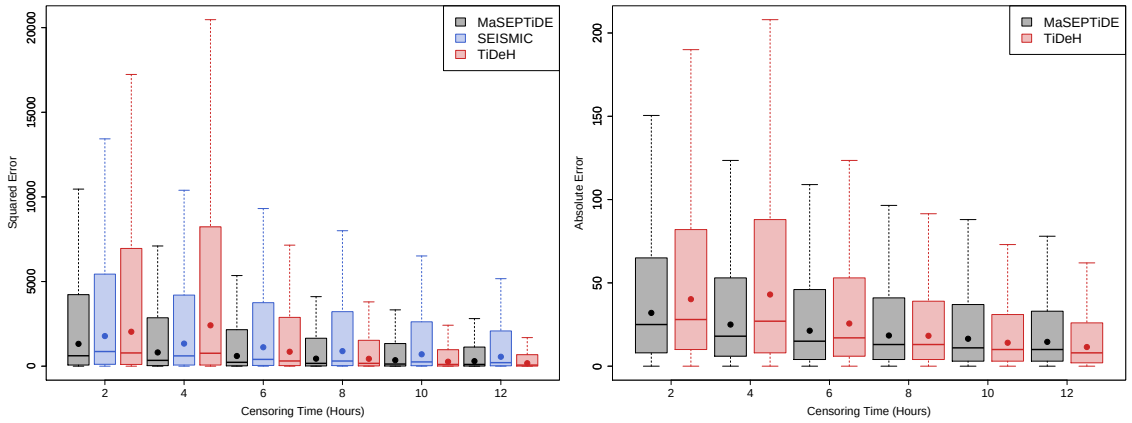

Figure 4.5.2: Left: squared prediction errors when the mean of the predictive distribution is used as the point prediction; Right: absolute prediction errors when the median is used. The thick horizontal bar in each boxplot shows the median of the errors, and the circular point shows the mean of the errors. Under both cases, the MaSEPTiDE model performs well at earlier censoring times $T = 2, 4, 6$ hours.

Table 4.5.4: Root mean squared errors (RMSEs) and mean absolute errors (MAEs) of predictions at different censoring times. Using the RMSE as the performance measure, the MaSEPTiDE model consistently outperforms the SEISMIC at all the censoring times, but only outperforms the TiDeH model at $T = 2, 4, 6$ hours. Using the MAE as the performance measure, the MaSEPTiDE model, similarly, outperforms the TiDeH model at times $T = 2, 4, 6$ hours.

T	RMSE			MAE	
(hours)	MaSEPTiDE	SEISMIC	TiDeH	MaSEPTiDE	TiDeH
2	36.3	42.2	45.1	32.0	40.2
4	28.5	36.5	49.1	25.0	43.0
6	24.5	33.4	29.2	21.3	25.6
8	21.1	29.8	20.9	18.4	18.2
10	18.8	26.5	16.3	16.5	14.1
12	17.3	23.6	13.2	14.6	11.5

To assess the performance of the conditional median predictions by different models, we shall use the mean absolute error (MAE), as advised by Gneiting (2011). The conditional median predictions by the MaSEPTiDE model were calculated by the simulation-based approach described in Section 4.4.3. The conditional median predictions by the TiDeH model were similarly calculated using a simulation-based approach, although the simulation of the TiDeH model was achieved by using a less efficient method where the events have to be simulated serially one after another using the rejective method of Lewis and Shedler (1979), as detailed in Section 3.4. The conditional median prediction by the SEISMIC has not been included in this comparison because this model does not specify the form of its intensity process beyond the censoring time, and thus we cannot use the simulation-based approach to obtain its conditional median.

The right panel of Figure 4.5.2 shows the absolute errors of the conditional median predictions by the MaSEPTiDE model and the TiDeH model at different censoring times, where, as before, the circular points indicate the MAEs of the predictions at the corresponding censoring times. See also Table 4.5.4 for the specific MAE values. By the MAE, the MaSEPTiDE model is superior to the TiDeH model at the censoring times $T = 2, 4, 6$ hours, and is comparable albeit slightly inferior at the larger censoring times $T = 8, 10, 12$ hours.

To further demonstrate how the MaSEPTiDE model outperforms the SEISMIC and the TiDeH model from a slightly different perspective, we append in Table B.2.1 the percentages of cascades with considerably small APE values ($< 5\%$), grouped according to the quantiles of the observed final popularity levels of these cascades, that is, $[q_{0.0}, q_{0.2})$,$[q_{0.2}, q_{0.4})$,$[q_{0.4}, q_{0.6})$,$[q_{0.6}, q_{0.8})$, and $[q_{0.8}, q_{1.0}]$. Consistent with the conclusions drawn based on the MdAPE, MAPE, RMSE, and MAE, the MaSEPTiDE model is highly accurate in predicting the popularity of tweets based on earlier censoring times $T = 2, 4, 6$ hours. It should be noted, however, that despite these highly accurate predictions, the MaSEPTiDE model may occasionally produce grossly erroneous prediction values. This issue will be discussed in Section 6.1.2.

By all the performance evaluation criteria considered, the prediction by the MaSEPTiDE model is clearly more accurate than those by the two competing models, especially when the prediction needs to be made based on a shorter censoring time, for example within six hours since the publication of the original tweet.

4.6 Discussion

Additional implementation details of the MaSEPTiDE model shall be given here, including the full summary statistics of a retweet cascade, the proposed approach to expedite simulations, the procedures to validate the estimation and prediction processes, and finally the various candidate models with different component functions.

4.6.1 Sample Summary Statistics

We have sampled a retweet cascade from the test set of the Twitter data presented in Section 1.3 to demonstrate the full summary statistics obtained from the procedures described in Section 4.1-4.4, and the summary statistics are presented in Table 4.6.1. As some of the interpretations of the statistics have been given prior to this point, we shall only highlight the remaining important ones.

The parameters in Table 4.6.1, or those of any other cascades in the test data set, were estimated based on a set of initial values learned from the training data, using the stochastic gradient descent method. The iterative method involves minimizing the objective function written as a sum of differentiable functions, or individual

Table 4.6.1: Summary statistics of a retweet cascade fitted by the MaSEPTiDE model at censoring times $T = 2, 4, \ldots, 12$ hours ordered according to the estimated parameters, the number of observed retweets, the integrated intensity, the p-value from the test of uniformity, the actual final popularity, predictions based on the conditional expectation and the conditional median with their corresponding performance measures, the prediction interval, and the integrated baseline intensity beyond the censoring time.

Statistics	Censoring time (hours)					
	2	4	6	8	10	12
$\hat{\alpha}$	95.957	13.165	100.646	100.656	98.246	99.447
$\hat{\beta}$	0.098	0.104	0.096	0.096	0.097	0.096
$\hat{\gamma}$	64.889	80.504	64.584	64.447	65.606	64.84
$\hat{\delta_1}$	1.349	1.317	1.315	1.313	1.326	1.318
$\hat{\delta_2}$	0.005	0.005	0.005	0.006	0.005	0.005
$N(T)$	1218	1347	1404	1443	1466	1490
$\hat{\Lambda}(T)$	1217.91	1348.07	1404.15	1443.06	1466.77	1490.24
p-value	0.19	0.17	0.24	0.25	0.26	0.27
$N(\tilde{T})$	1668	1668	1668	1668	1668	1668
STE mean[a]	1645.41	1698.63	1693.71	1695.91	1681.46	1690.93
SB mean[b]	1637.44	1644.44	1699.80	1706.34	1690.64	1697.00
SB median[c]	1646	1699	1694	1695	1682	1690
APE	1.35	1.84	1.54	1.67	0.81	1.37
Squared error	933.91	555.07	1011.24	1469.96	512.57	841.00
Absolute error	22	31	26	27	14	22
$q_{0.025}$	1218	1347	1404	1443	1466	1490
$q_{0.975}$	2288	2051	2129	2073	2006	2042
$\hat{N}$	427.06	351.63	289.71	252.91	215.46	200.93

[a]Conditional mean prediction from the solve-the-equation approach
[b]Conditional mean prediction from the simulation-based approach
[c]Conditional median prediction from the simulation-based approach

negative log-likelihood functions, with the aim to obtain a parameter vector which has a considerably small error based on a certain learning rate and a set batch size. More details on the method can be found in Saad (1998) and Kiwiel (2001).

The insights of retweet activities can also be gained based on the estimated parameter values in a similar fashion to those provided in Section 4.5.1, except that we now have censoring times $T = 2, 4, \ldots, 12$ hours instead of a fixed $\tilde{T} = 7$ days. Furthermore, if we apply the estimated parameters $\hat{\theta}$ to the integrated intensity from 0 to T, that is,

$$\hat{\Lambda}(T) = \Lambda(T; \hat{\theta}) = \int_0^T \lambda(s; \hat{\theta}) \, \mathrm{d}s, \tag{4.6.1}$$

for example using $T = 2$ hours and $\hat{\theta} = (95.957, 0.098, 64.889, 1.349, 0.005)^\top$, we would obtain a value consistent to that of the observed popularity up to the censoring time, at $N(T) = 1218$ and $\hat{\Lambda}(T) = 1217.91$. Similar procedures can be applied to

later censoring times to yield the respective values of $N(T)$ and $\hat{\Lambda}(T)$ in Table 4.6.1, where marginal differences can be observed. While this procedure seems to have limited utility here, it is important when we setup the prior distribution for model parameters in Chapter 5, or more specifically, Section 5.2.3.

Next, the p-values obtained using the Kolmogorov-Smirnov test of uniformity shown in Table 4.6.1 are sufficiently larger than 0.05 at all the censoring times, implying that the residuals are uniformly distributed. An informal approach to evaluate the goodness-of-fit for a point process model, as discussed in Section 4.3, would be to visually inspect the uniformity of its residuals, using for example the histogram. Figure 4.6.1 shows the histograms of residuals at different censoring times $T = 2, 4, \ldots, 12$ hours. Purely visually, the residuals seem to be rather uniformly distributed.

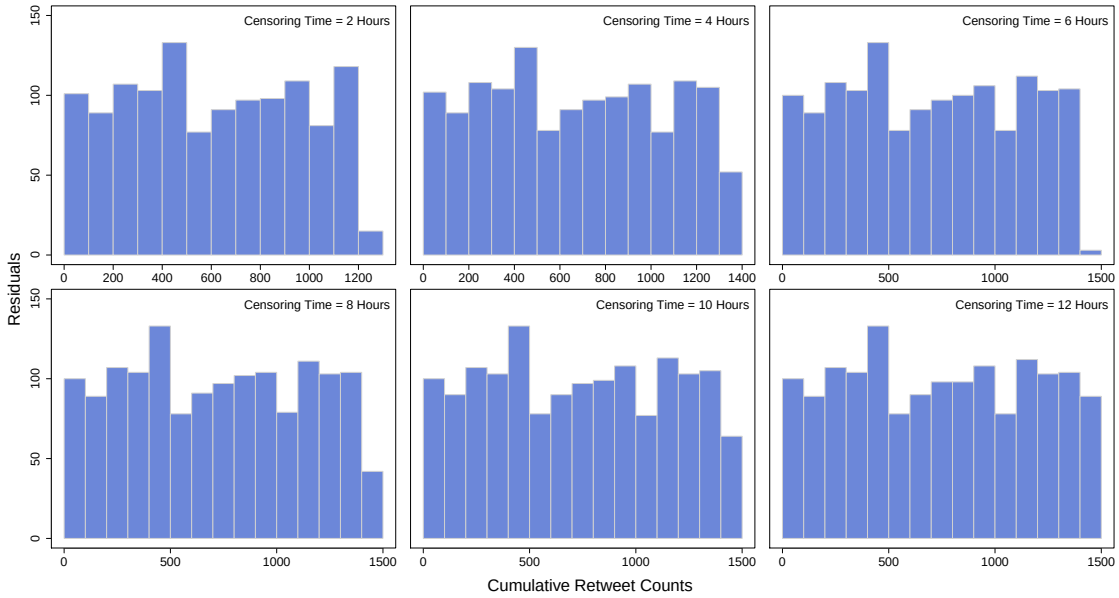

Figure 4.6.1: Histograms to visualize the uniformity of residuals for the MaSEPTiDE model with parameters set to their ML estimates $\hat{\theta}$ at different censoring times. The x-axes indicate the cumulative retweet counts, while the y-axes indicate the point process residuals $\hat{\lambda}(\tau_i), i = 1, 2, \ldots, N(T)$ over the interval $(0, \hat{\lambda}(T)]$. The residuals at all the censoring times seem to be rather uniformly distributed.

We have also mentioned that point prediction based on the conditional expectation can be obtained by using either the solve-the-equation approach or the simulation-based approach, and that with sufficiently many simulation replications, both methods should yield consistent prediction values. This is proven in Table 4.6.1, where point predictions based on both approaches seem to exhibit only marginal differences at all the censoring times $T = 2, 4, \ldots, 12$ hours.

Using the final popularity of the retweet cascade $N(\tilde{T})$, we can obtain different performance measures based on different functionals of the predictive distribution at each censoring time. As mentioned in Section 4.5.3, we have used the APE as

a performance measure for the conditional expectation from the solve-the-equation approach, and the squared error or absolute error for the conditional expectation or the conditional median from the simulation-based approach. It is worth noting that although the MdAPE or MAPE is theoretically inconsistent with the conditional expectation used here, the final conclusions drawn based on different performance measures are largely similar. From Table 4.6.1, all the performance measures demonstrate that the MaSEPTiDE model is able to predict the future popularity at each censoring time reasonably well, and the precision increases when more data accumulates over time.

The prediction interval at each censoring time can also be obtained from the simulated event numbers by using some appropriate quantiles, for example $[q_{0.025}, q_{0.975}]$, with $q_{0.025}$ denoting the 0.025-quantile and $q_{0.975}$ denoting the 0.975-quantile. The future popularity is considered to have been correctly predicted when the prediction interval at a certain censoring time includes the actual final popularity, and that the interval is of plausible range. A close inspection on Table 4.6.1 reveals that the prediction interval has successfully covered the actual final popularity since $T = 2$ hours, and the interval gets narrower over time, implying a gradual increase in precision.

Figure 4.6.2 shows the simulated sample paths and the corresponding prediction interval for this specific retweet cascade censored at $T = 2$ hours. The close

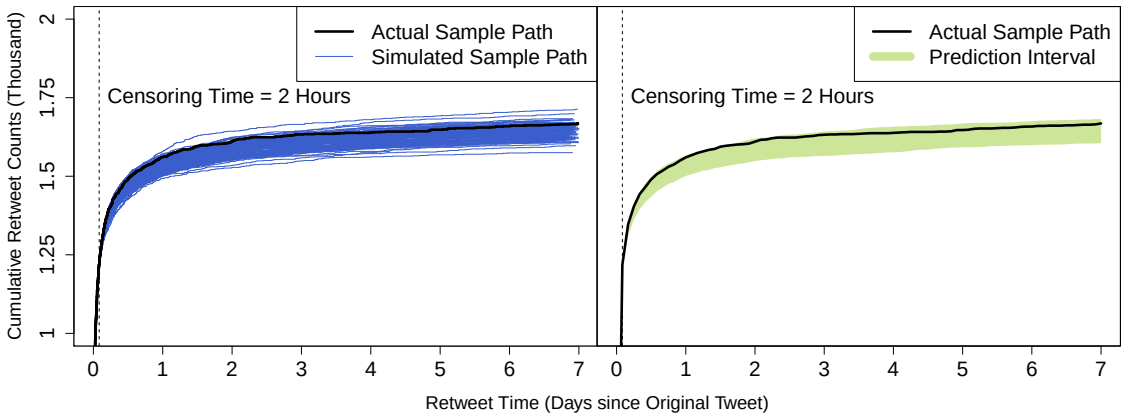

Figure 4.6.2: Left: sample paths generated using the simulation-based approach for the MaSEPTiDE $\tilde{N}$ process at $T = 2$ hours; Right: the corresponding prediction interval obtained based on the quantiles $[q_{0.025}, q_{0.975}]$ from the generated event numbers. The plots in both panels indicate that the future popularity of the tweet has been successfully predicted.

proximity of the simulated trajectories to the actual sample path as shown in the left panel of Figure 4.6.2 demonstrates the capability of our MaSEPTiDE model to quickly capture the retweeting dynamics and predict the future evolution of this cascade. The right panel of Figure 4.6.2 shows the corresponding prediction interval,

and conveys the similar message that accurate tweet popularity prediction has been made based on early retweeting dynamics.

When simulating the MaSEPTiDE $\tilde{N}$ process over the interval $(0, \tilde{T} - T]$, we note that the procedure can be very time consuming when the number of events to be simulated on each iteration is large. For that, it would be useful to estimate the minimum number of events to be generated, which can be calculated by integrating over the baseline intensity function $\tilde{\nu}(\cdot)$ for the MaSEPTiDE $\tilde{N}$ process from 0 to $\tilde{T} - T$ with the estimated parameters $\hat{\theta}$, or specifically,

$$\hat{N} \equiv \hat{N}(\tilde{T} - T) = N(\tilde{T} - T; \hat{\theta}) = \int_0^{\tilde{T}-T} \tilde{\nu}(s; \hat{\theta}) \, \mathrm{d}s. \tag{4.6.2}$$

The $\hat{N}$ value naturally decreases as the censoring time increases, as shown in Table 4.6.1. This value can be used to expedite the simulation of the MaSEPTiDE $\tilde{N}$ process, as demonstrated in Section 4.6.2.

4.6.2 Expediting Simulation

When using the simulation-based approach, we note that, for some very popular tweets, the retweet cascades are very long and the numbers of retweet events to be simulated are very large, and therefore simulations can take a long time to complete. A trick we used to mitigate this issue is to simulate the process $\tilde{N}$ with a smaller baseline intensity function, say $\tilde{\nu}(\cdot)/S$, and inflate the simulated event numbers by the factor S. This factor can be referred to as the acceleration factor.

The value $\hat{N}$ from (4.6.2) can help to determine the acceleration factor suitable to be used in the simulations. Under some circumstances, one may simply resort to using the $\hat{N}$ value as the acceleration factor in place of S. However, this may be be problematic for some retweet cascades as the issue of overdeflating the baseline intensity may arise. Specifically, if the baseline intensity has been deflated disproportionately, then there will be barely any events generated, and the numbers of simulated events will be zeros prior to inflating them. This contradicts to the reality when some events are actually simulated if the baseline intensity has not been deflated. Through several numerical experiments, we hereby propose the solution for how the acceleration factor S can be determined based on $\hat{N}$, as shown in Table 4.6.2. Such treatment is reasonably effective to improve the efficiency of the MaSEPTiDE simulations, without disrupting the conditional mean and conditional median predictions based on the simulated event numbers compared to when the baseline intensity is kept at its original form.

As an illustrative example, we take the retweet cascade shown in Table 4.6.1, and select the censoring time $T = 2$ hours, with $\hat{N} = 427.06$. Using the deflated

Table 4.6.2: The acceleration factor S used to expedite the simulations of the MaSEPTiDE $\tilde{N}$ process, based on the range of $\hat{N}$, where $\lfloor x \rceil$ denotes the nearest integer of x. The intuition is to speed up the simulations without overdeflating the baseline intensity function $\tilde{\nu}(\cdot)$.

$\hat{N}$	$[0, 10)$	$[10, 10^2)$	$[10^2, 10^3)$	$[10^3, 10^4)$	$[10^4, \infty)$
S	1	$\lfloor 0.25\hat{N} \rceil$	$\lfloor 0.50\hat{N} \rceil$	$\lfloor 0.75\hat{N} \rceil$	$\lfloor \hat{N} \rceil$

baseline intensity $\tilde{\nu}(\cdot)/S$ where $S = \lfloor 0.5(427.06) \rceil = 214$ based on Table 4.6.2, with the estimated parameter values for $\alpha, \beta, \gamma, \delta_1$, and δ_2 being 95.957, 0.098, 64.889, 1.349, and 0.005 respectively, the simulated event numbers for the MaSEPTiDE $\tilde{N}$ process at 100 replications are shown as follows. These event numbers are then

0 0 0 1 1 6 2 2 4 5 0 1 3 1 2 1 2 3 1 8 1 2 2 3 2
0 2 2 1 1 3 5 2 1 2 3 2 1 1 2 2 2 4 3 5 0 4 3 3 2
1 1 3 5 3 0 3 0 1 3 1 1 2 1 2 1 2 1 3 1 0 4 1 2 1
3 2 1 2 4 3 0 0 1 2 3 0 2 0 2 3 4 2 1 2 2 1 0 2 2

inflated with the acceleration factor S, and added to the observed number of events $N(T) = 1218$. The vector of event counts can then be used to obtain the conditional expectation and the conditional median, at 1637.44 and 1646 respectively. The prediction interval can also be obtained from the same set of simulated event numbers, where $[q_{0.025}, q_{0.975}] = [1218, 2288]$.

A natural question to ask is whether or not the remarkable performance based on interval prediction applies to the remainder of the retweet cascades in the data set. That said, using the whole test data, the coverage probabilities at the stipulated censoring times based on $[q_{0.025}, q_{0.975}]$ are all less than the nominal coverage probability of 95%, ranging only from 59.5% when $T = 2$ hours to 65.9% when $T = 12$ hours. Ideally, when the process generating mechanism has been properly identified and captured, the coverage probability should be very close to the nominal level (Brooks and Gelman, 1998), even when the cascade has only been observed for a short period of time. This will be substantiated by the systematic validation procedures in Section 4.6.3.

4.6.3 Simulation Experiments

In order to validate the MaSEPTiDE estimation and prediction procedures, we utilized the Poisson cluster process interpretation demonstrated in Section 4.1.2. That is, we generated the immigrants based on the baseline intensity function $\nu(\cdot)$ and offspring using the excitation function $\omega(\cdot, \cdot, \cdot)$, over the time interval $(0, \tilde{T}]$. The corresponding event mark for each generated event time was simulated based on a common normal density $f(\cdot; \mu, \sigma^2)$ with its mean and variance estimated from

the training data set. We generated 1,000 synthetic cascades by this convention.

For each of the synthetically generated retweet cascades, we censored them accordingly at $T = 2, 4, \ldots, 12$ hours. Then, we used the simulation-based approach to obtain the corresponding prediction interval $[q_{0.025}, q_{0.975}]$ at each of the censoring time for each of the cascade, based on the MaSEPTiDE $\tilde{N}$ process. It should be noted, however, that the procedures used here to obtain the prediction interval vary slightly than those discussed in Section 4.4.3. Specifically, for a retweet cascade at a certain censoring time T, instead of using the same set of parameters $\hat{\theta}$ to simulate the MaSEPTiDE $\tilde{N}$ process, we obtained the Hessian matrix H from the estimated parameter values $\hat{\theta}$ and used it to generate new parameter vectors θ_k^* following a multivariate normal distribution, based on the mean vector $\hat{\theta}$ and the covariance matrix H^{-1}, at k simulation replications. Intuitively, the matrix H should be positive-definite to be invertible, and the parameter values should naturally follow that $\alpha > 0, \beta \geqslant 0, \gamma \geqslant 0, \delta_1 > 1$, and $\delta_2 > 0$ as shown in Section 4.1.1.

The newly acquired parameter vectors θ_k^* can then be used to simulate the MaSEPTiDE $\tilde{N}$ process for each of the synthetic retweet cascades, thus yielding their respective prediction intervals at each of the censoring times. Table 4.6.3 shows the coverage probabilities of interval predictions at different censoring times based on all the synthetic retweet cascades. It can be observed that approximately

Table 4.6.3: The coverage probabilities of interval predictions based on the synthetic data censored at times $T = 2, 4, \ldots, 12$ hours. The coverage gradually increases with time, attaining the nominal level at around $T = 8$ hours.

Censoring time (hours)	2	4	6	8	10	12
Coverage (%)	85.8	92.0	93.3	94.6	94.7	95.9

86% of the final popularity has been successfully predicted from as early as $T = 2$ hours, and the coverage increases over time, achieving the nominal probability of 95% at around $T = 8$ hours.

4.6.4 Candidate Models

Prior to arriving at the final form of our MaSEPTiDE intensity function, we have also considered other combinations of the component functions, illustrated as follows,

$$
\begin{aligned}
p_1(t;\beta) &= e^{-\beta t} & \phi_1(t;\delta) &= \delta e^{-\delta t} \\
p_2(t;\beta) &= \left(1 + \frac{t}{\beta_1}\right)^{-\beta_2} & \phi_2(t;\delta) &= \frac{\delta_2(\delta_1 - 1)}{\delta_1}\left(1 + \frac{\delta_2 t}{\delta_1}\right)^{-\delta_1} \\
p_3(t;\beta) &= 1 \wedge \left(\frac{t}{\beta_1}\right)^{-\beta_2} & \phi_3(t;\delta) &= \frac{\delta_2 - 1}{\delta_1\delta_2}\left(1 \wedge \left(\frac{t}{\delta_1}\right)^{-\delta_2}\right)
\end{aligned}
$$

where the forms of $p(\cdot)$ and $\phi(\cdot)$ may vary but the impact function $r(\cdot)$ is fixed. This means that the excitation function may be a combination of $p_1(\cdot)\phi_1(\cdot), p_1(\cdot)\phi_2(\cdot), \ldots,$ and $p_3(\cdot)\phi_3(\cdot)$, with each of them multiplied with the impact function $r(\cdot)$. The specific combination of the component functions reported and used in our model formulation that has the best fit to the training data is $\omega(\cdot,\cdot,\cdot) = p_1(\cdot)r(\cdot)\phi_2(\cdot)$, as shown in (4.1.3).

4.7 Concluding Remarks

We have proposed a marked self-exciting point process model in this chapter, termed the MaSEPTiDE, to model the retweeting dynamics and to predict the future popularity of tweets. The MaSEPTiDE is capable of modelling a large number of retweet cascades adequately, and its prediction performance is superior to those of the competing models and approaches in the literature that require the same input.

When the prediction is based on observing a retweet cascade for a long period of time, the approach based on the TiDeH model of Kobayashi and Lambiotte (2016) is found to outperform our model by a small margin. However, considering the fact that this small advantage of the TiDeH model is not realized until the retweet cascade has been observed for eight hours or longer, when the majority of the retweet events would have already occurred, its practical significance is rather limited. On the contrary, the approach based on the MaSEPTiDE model is able to provide accurate prediction of the final popularity based on observations within two hours since the publication of the original tweet. Another issue with the TiDeH model is that the nonparametric estimation step to obtain the initial raw estimate of the infectivity curve needs a large amount of data to work well. In fact, in their numerical experimentation, Kobayashi and Lambiotte (2016) only verified the superior performance of their prediction approach relative to the SEISMIC on 738 very long cascades (containing 2,000 or more retweets), which account for less than 0.5% of all the retweet cascades. In contrast, the approach based on the MaSEPTiDE is fully parametric, and therefore does not require as much data to estimate.

The specific parametric forms of the functions in the MaSEPTiDE model have been selected from a class of candidate models by comparing their goodness-of-fit on the retweet cascades in the training data set and identifying the model that can fit most of the cascades. In the class of candidate models, we have considered other parametric forms of the component functions, such as infectivity functions that decay at polynomial rate, and memory kernel functions that decay exponentially fast. The model with the specific forms of the component functions reported herein has the best goodness-of-fit on the training data set.

To further improve the MaSEPTiDE model, more complex models, such as those

that incorporate the calendar time effects (Fox et al., 2016; Kobayashi and Lambiotte, 2016) are worth considering. Another aspect of our approach that can be improved is that our approach still requires the observation of the retweet cascade for a substantial amount of time to accumulate enough data to identify the model (a post-publication prediction method), even though the required observation time is much less compared to approaches based on other models such as the TiDeH model. If we make stronger assumptions on the model parameters across the cascades, then parameter estimation might be achieved using only the training data set, which in turn allows us to predict the final popularity of a tweet as soon as it is published, or even before it is published.

On another remark, the MaSEPTiDE model is a microscopic level prediction method. This implies that the retweet cascades can be evaluated individually, from parameter estimation to point prediction using the solve-the-equation approach or the simulation-based approach. An intuitive way to obtain the results for a huge number of cascades efficiently under the microscopic level method would be to run them in parallel, using for example some typical computational clusters.

Finally, an important limitation of the data considered in our work, originally collected by Zhao et al. (2015), is that it contains only cascades with at least 49 retweets. Such data is by no means representative of all the tweets published by Twitter users on the network, as the majority of the tweets do not even get a single retweet. Therefore, models developed based on such data are only useful for popularity predictions of reasonably popular tweets. To develop models suitable for the predictions of the popularity of average tweets, one would need to collect suitable random samples of tweets and their retweet cascades, and build models accordingly.

Chapter 5

An Empirical Bayes Approach

Tweets are renowned for their ephemeral nature, gaining popularity rapidly but faced with their eventual decays soon afterwards. Such episode begs emphasis on earlier tweet popularity predictions, or popularity predictions of tweets based on shorter censoring times. The MaSEPTiDE model described in Chapter 4 has served this purpose well, demonstrating superior prediction performance over the competing approaches in the literature. Nonetheless, it is still a post-publication prediction method which needs to rely on some observations of retweets prior to making a prediction. This affirms that pre-publication tweet popularity prediction remains a gap to be bridged.

To the best of our knowledge, there has been no published work that specifically addresses the problem of pre-publication tweet popularity prediction, except for that of Martin et al. (2016), which has been discussed in Section 3.2.1. Performing tweet popularity prediction at the time of publication is thus one of the motivations to our work in this chapter. For that, it is worth noting that information readily available at time zero for each retweet cascade we consider comprises only of t^0 and n^0, which correspond to the relative tweet time and the number of followers of the tweeter respectively. Our novel prediction methodology proposed herein shall leverage such information.

Another important motivation to this chapter is the limited use of the training data set by the state-of-the-art approaches for post-publication tweet popularity predictions. For instance, the MaSEPTiDE model we proposed in Chapter 4 which was found to outperform the selected competing prediction methods, only utilizes the training data to inform its model construction, and completely ignores the training data when fitting the model to the observed retweet sequence of a specific tweet whose popularity is to be predicted. This has led to computational difficulties when the tweet in question has only accumulated a small number of retweets by the censoring time. As a result, the approach may struggle when attempting to obtain the model parameter estimate and may subsequently fail to produce any prediction

at all.

The last motivation is the computational complexity of the approaches based on the MaSEPTiDE model and the TiDeH model. When making tweet popularity predictions from the estimated models, each of the approaches requires either numerically solving an integral equation for a positive function followed by numerically integrating the solution function over a suitable interval, or simulating the fitted point process model from the censoring time to a future time point over sufficiently many iterations followed by extracting suitable numerical features from the simulated sample paths. However, the numerical procedure to solve the integral equations can occasionally fail to produce a legitimate solution, the numerical integration may fail even though a legitimate solution is obtained, and the simulation-based method can be intolerably slow, especially on very popular tweets.

In this chapter, we propose a novel approach which is not only capable of making pre-publication tweet popularity prediction, but is also able to produce more accurate predictions than the competing approaches at later censoring times. The approach is based on an inhomogeneous Poisson process model for the retweet time sequence, with advantages such as the ease-of-implementation and computational stability. Therefore, it is also much simpler compared to the self-exciting point process models in the literature (Zhao et al., 2015; Kobayashi and Lambiotte, 2016; Mishra et al., 2016), and is straightforward to make a prediction after fitting the model.

We further propose a novel empirical Bayes type approach where the prior distribution for the model parameters specific to a tweet is constructed using the training data external to the tweet, and the maximizer of the posterior density function is taken as the estimator of the tweet specific parameters. Moreover, the approach enables prediction at time zero, by utilizing the estimated model based on taking the maximizer of the prior density function as the estimator of the tweet specific model parameters. The incorporation of external knowledge through the prior distribution not only enables pre-publication tweet popularity predictions, but also leads to more stable estimates of the tweet specific parameters than the ML estimator, and more accurate popularity predictions at various censoring times overall.

The remainder of this chapter is systematically arranged as follows. First, we present the proposed model for a retweet sequence in Section 5.1, followed by describing how the knowledge internal and external to the retweet sequence can be combined using an empirical Bayes type approach in Section 5.2. Then, we explain how predictions can be made using suitable functionals of the predictive distribution for the number of retweets implied by the fitted model in Section 5.3. The main results are exhibited in Section 5.4, and further elaborations on the proposed methodology can be found in Section 5.5. Finally, the concluding remarks are given

in Section 5.6.

5.1 Model Formulation

Recall that for a retweet cascade, the retweet times are given by $\tau_1 < \tau_2 < \ldots$, relative to the posting time t^0 of the original tweet and that n^0 denotes the number of followers of the original tweeter. We model the sequence of retweet times by a Poisson process with a time-dependent intensity function $\lambda(t)$. Specifically, let $N(t) = \#\{i \geqslant 1 : \tau_i \leqslant t\}$ count the number of retweets up to time t. Then $N(t)$, $t > 0$, is assumed to be a Poisson process with $\mathbb{E}[N(t) - N(s)] = \int_s^t \lambda(u)\,\mathrm{d}u$, for $0 \leqslant s < t$. The intensity function $\lambda(t)$ is assumed to take the following multiplicative form,

$$\lambda(t) = p(t)d(t), \tag{5.1.1}$$

where the function $p(t)$ shall reflect the ageing effect of the original tweet on its retweet intensity at time t, and can be similarly referred to as the *infectivity function*. As the older a tweet is, the less likely it will get retweeted, the function $p(\cdot)$ should be decreasing. Following the literature (Malmgren et al., 2008), we assume it decreases at polynomial rate, where

$$p(t) = \alpha(1 + \beta t)^{-\gamma}, \tag{5.1.2}$$

for parameters $\alpha > 0$, $\beta > 0$, and $\gamma > 0$. The parameters α, β, and γ are referred respectively to as the magnitude parameter, the scale parameter, and the shape parameter. These parameters are assumed to be tweet specific, and may be different for retweet cascades originating from different tweets.

The nonnegative function $d(\cdot)$ is a global parameter common to all retweet cascades which reflects the circadian rhythm of all Twitter users, and is naturally assumed to be periodic with period one day. Specifically, if we measure time in unit days, then there is a function $\rho(\cdot) \geqslant 0$ such that

$$d(t) = \rho(t^0 + t - \lfloor t^0 + t \rfloor), \tag{5.1.3}$$

where $\lfloor x \rfloor$ indicates the greatest integer $\leqslant x$. Here, $d(\cdot)$ is assumed to be a smooth function, but is otherwise unspecified. For identifiability, we assume that the function $\rho(\cdot)$ integrates to unity, so that it is a probability density function supported by $[0, 1)$. The smoothness and periodicity of the function $d(\cdot)$ also imply that the function $\rho(\cdot)$ is smooth, and furthermore satisfies the continuity condition,

$$\rho(0) = \lim_{t \downarrow 0} \rho(t) = \lim_{t \uparrow 1} \rho(t). \tag{5.1.4}$$

For convenience, the function $\rho(\cdot)$ shall be referred to as the *rhythm function*, and its

role in portraying the circadian rhythm of Twitter users shall be gradually demonstrated as we manoeuvre through the rest of this chapter. A somewhat relevant work modelling the information diffusion of retweet cascades based on the inhomogeneous Poisson process has also been done recently by Lee and Wilkinson (2018).

5.2 Parameter Estimation

Before we can combine the knowledge internal and external to a specific retweet time sequence, the function $\rho(\cdot)$ in (5.1.3) and the parameters in (5.1.2) have to be estimated first, detailed respectively in Section 5.2.1 and Section 5.2.2. The proposed novel approach to combine the knowledge is subsequently described in Section 5.2.3.

5.2.1 Estimation of the Rhythm Function

Our assumption that all retweet cascades share the same circadian rhythm function $\rho(\cdot)$ amounts to assuming that this function is equal to the density function of the distribution of the tweet publication times in the interval $[0, 1)$, where 0 and 1 stand respectively for the beginning and the end of the day. Therefore, we can estimate $\rho(\cdot)$ nonparametrically using the *kernel density estimator* (KDE; see Silverman, 1986, Section 2.4) with the publication times of the original tweets in the training data.

To correct for the well-known boundary effects suffered by the KDE, and to ensure the continuity condition in (5.1.4), we have adopted a pseudodata approach which is similar in spirit to the data reflection approach discussed in Silverman (1986) and the pseudodata approach of Cowling and Hall (1996). Specifically, if $t_1^0, t_2^0, \ldots, t_n^0 \in [0, 1)$ denote the data, that is, the publication times of the original tweets measured in days since 00:00:00 on the dates they were posted, we augment the data by adding $t_1^0 - 1, \ldots, t_n^0 - 1$ and $t_1^0 + 1, \ldots, t_n^0 + 1$. Following this, we estimate the density on $[0, 1)$ using the KDE with the augmented data, and subsequently rescale the estimates so that the estimated density curve $\hat{\rho}(\cdot)$ integrates to unity. Finally, the estimated rhythm function for a retweet sequence originating from time t^0 is simply

$$\hat{d}(t) = \hat{\rho}(t^0 + t - \lfloor t^0 + t \rfloor). \tag{5.2.1}$$

In our numerical implementation of the KDE, we have used the function `density` from the `stats` package of R (R Core Team, 2016), with the biweight kernel $K(x) = 15/16(1 - x^2)_+^2$ and the bandwidth parameter selected using the default normal reference distribution approach based on the unaugmented data.

5.2.2 Estimation of the Infectivity Function

The parameters for a specific retweet cascade can be estimated based on the ML approach discussed in Section 2.5, which involves maximizing the likelihood of the parameters, or equivalently its logarithm, relative to the observed retweet times up to the censoring time T, as a function of the parameters. Thus, our main objective here is to estimate the parameter vector $\theta = (\alpha, \beta, \gamma)^\top$ for each individual retweet cascade.

The log-likelihood function specific to our model over the interval $[0, T]$ takes the following form,

$$\ell(\theta, d) = \sum_{i=1}^{N(T)} \log \lambda(\tau_i; \theta, d) - \int_0^T \lambda(t; \theta, d)\, \mathrm{d}t, \tag{5.2.2}$$

where $\lambda(t; \theta, d) = p(t; \theta)d(t) = \alpha(1 + \beta t)^{-\gamma} d(t)$, and the function $d(\cdot)$ is fixed at its estimate $\hat{d}(\cdot)$ when $\ell(\theta, d)$ is optimized to estimate θ. This implies that the ML estimation of the parameters θ shown here is based on maximizing the logarithm of the likelihood function which depends solely on the component function $p(\cdot)$.

The ML approach to estimate the parameters θ described above only uses the retweet history of the specific tweet for which the prediction of its final popularity is desired. The resulting estimate of the parameters and the prediction of its final popularity based on these estimated parameters can be very unreliable, or even not available at all when the retweet sequence is observed for too short a period of time before any retweets can occur. To overcome this issue, we shall incorporate prior knowledge learned from the training data set into the estimation of θ, using a novel empirical Bayes type approach, described in Section 5.2.3. Further reading on the empirical Bayes methods can be found in the works of Morris (1983) and Casella (1985).

5.2.3 An Empirical Bayes Approach

The parameter estimates based on the empirical Bayes type approach, or the EB estimates in short, require the acquisitions of the ML estimates from the training data set. Therefore, as a first step, we compute the ML estimate for each of the complete retweet sequences in the training data described in Section 1.3, and denote these estimates by $\hat{\theta}_i^0 = (\hat{\alpha}_i^0, \hat{\beta}_i^0, \hat{\gamma}_i^0)$, for $i = 1, 2, \ldots, 71815$.

In the second step, we fit three separate nonparametric regression models with $y_i = \log \hat{\alpha}_i^0$, $\log \hat{\beta}_i^0$ and $\log \hat{\gamma}_i^0$ as the respective response variables, and $x_i = (m_i^0, t_i^0)$ as the input variables, where $m_i^0 = \log(n_i^0 + 1)$, using the locally weighted kernel regression approach (LOESS; Cleveland and Devlin, 1988; Fan, 2018). In our nu-

merical implementation of the nonparametric regression, we have used the `loess` function from the `stats` package of R, with the degree of the local polynomial set to the default value of 2, the kernel function set to the default tricubic kernel $K(x) = \frac{70}{81}(1 - |x|^3)_+^3$, and the respective span parameters selected using the *generalized cross validation* (GCV; Golub et al., 1979) method.

In the third step, for a tweet posted at time t^0 by a tweeter with n^0 followers, we predict the values of the log-transformed parameters $\eta = (\eta_1, \eta_2, \eta_3) \equiv (\log\alpha, \log\beta, \log\gamma)$ using the nonparametric regression models obtained in the last step, with $x = (m^0, t^0)$ as the input. Then, we denote the predicted log-parameter values by $\tilde{\eta}^0 = (\tilde{\eta}_1^0, \tilde{\eta}_2^0, \tilde{\eta}_3^0) \equiv (\log\tilde{\alpha}^0, \log\tilde{\beta}^0, \log\tilde{\gamma}^0)$, and the associated standard errors by (e_1, e_2, e_3), both of which are obtainable from the `predict.loess` function in R.

Next in the fourth step, we define a *prior density function* for the log-parameters η as follows,

$$\pi(\eta) = f(\eta_1; \tilde{\eta}_1^0, e_1^2) f(\eta_2; \tilde{\eta}_2^0, e_2^2) f(\eta_3; \tilde{\eta}_3^0, e_3^2), \tag{5.2.3}$$

with $f(\cdot; \mu, \sigma^2)$ denoting the normal density function with mean μ and variance σ^2, so that $\tilde{\eta}^0$ comes as the maximizer of $\pi(\eta)$. Here, we note that the *prior distributions* for the log-parameters $\eta = (\eta_1, \eta_2, \eta_3)$ are the respective *confidence distributions* (Xie and Singh, 2013) based on the training data for their means $\mathbb{E}[\eta_i]$, $i = 1, 2, 3$, when they are treated as random variables with means depending on (m^0, t^0).

Finally, we define our estimator at censoring time T for the parameters θ as $\tilde{\theta} = e^{\tilde{\eta}}$, where $\tilde{\eta}$ is the maximizer of the following *criterion function*,

$$\tilde{\ell}(\eta) = \log\pi(\eta) + \ell(e^{\eta}, \hat{d}), \tag{5.2.4}$$

with $\ell(\cdot, \cdot)$ given in (5.2.2) and $\hat{d}(\cdot)$ given in (5.2.1). In fact, if we treat $d = \hat{d}$ as known, then $\tilde{\ell}(\eta)$ is, up to an additive constant, equal to the logarithm of the *posterior density* of η. Therefore, $\tilde{\eta}$ can be regarded as a *maximum a posteriori (MAP)* estimator of η. Since our construction of the prior density for η is suggested by the data, the estimation approach might be considered an *empirical Bayes (EB)* approach, despite its obvious difference in the construction of the prior distribution for model parameters than that in the conventional empirical Bayes estimation approach, described for example in Section 1.2 of Efron (2010).

For convenience, the steps involved in obtaining the EB estimates have been summarized in Figure 5.2.1. We note that the final criterion function has essentially incorporated knowledge internal and external of a specific retweet sequence, coming respectively from the retweet sequence itself and the training data. It is also noteworthy that at censoring time zero, the maximizer of the prior density function, namely $\tilde{\eta}^0$, will be taken as the estimator of the tweet specific model parameters,

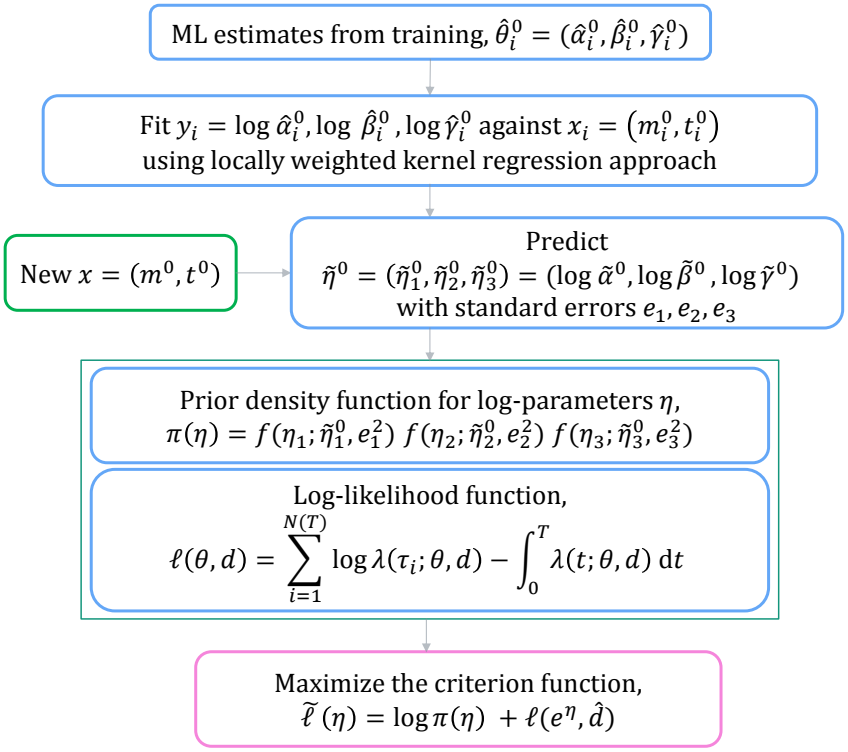

Figure 5.2.1: A summary of the procedures involved to obtain the empirical Bayes estimates. The final criterion function combines the knowledge internal and external to a retweet cascade, depending respectively on the current log-likelihood function and the log-prior density function. When the censoring time is at zero, the maximizer of the prior density function $\tilde{\eta}^0$ will be taken as the estimator of the tweet specific model parameters.

which enables pre-publication tweet popularity prediction.

The most time-consuming calculation in the above steps is the fitting of the model on all the retweet cascades in the training data, but this can be done in parallel using a large number of CPUs available on a typical computational cluster. Moreover, this time-consuming step, as well as the other relatively time-consuming steps such as the selection of the GCV smoothing parameters, only need to be performed once to obtain the prior distribution of the tweet specific parameters. Once the prior distribution is constructed, it is to be used in the MAP estimation of the tweet specific parameters for all the tweets in the test data set at all the censoring times of interest.

At this point we shall emphasize the advantage of using the Poisson model in making tweet popularity predictions, which is the ease of its likelihood evaluation that only requires linear time in the number of retweets. In contrast, the likelihood computations for the various self-exciting models in the literature, such as those of Zhao et al. (2015), Kobayashi and Lambiotte (2016), and Mishra et al. (2016), all require quadratic time due to the dependence of the intensity process on all previous

instances of retweet times for a specific retweet sequence in each of the models.

It is also worth noting that, although we have described the EB estimation approach assuming a Poisson process model, it is obvious that the approach to incorporate both the internal history of a specific retweet sequence and the external knowledge on other similar retweet sequences into parameter estimation is also applicable with other point process models, such as the MaSEPTiDE model and the TiDeH model. Based on the similar acronym, these models with the incorporation of prior knowledge can be referred to as the EB MaSEPTiDE model and the EB TiDeH model, and shall be described in Section 6.1 and Section 6.2 respectively.

5.3 Predicting the Popularity

After the parameters are estimated, the model for a specific retweet sequence is identified. We might then proceed to using the mean or median of the predictive distribution of the number of retweets from a censoring time T to a future time point $\tilde{T}$ implied by the identified model, plus the number of retweets observed by time T, as a point prediction of the total number of retweets by time $\tilde{T}$, as shown in (2.8.1).

For the Twitter data set considered in Section 1.3, since we know a priori that the final popularity is at least 49, the mean and median of the distribution for the number of future retweets should be calculated conditional on $N(\tilde{T}) - N(T) \geqslant 49 - N(T)$. Under the Poisson process model, $N(\tilde{T}) - N(T)$ is Poisson distributed with its mean equals to the integral of the identified intensity function from T to $\tilde{T}$, or equivalently its shifted intensity from 0 to $\tilde{T} - T$,

$$\int_T^{\tilde{T}} \lambda(t; \tilde{\theta}, \hat{d})\, \mathrm{d}t \equiv \int_0^{\tilde{T}-T} \tilde{\lambda}(t; \tilde{\theta}, \hat{d})\, \mathrm{d}t,$$

where $\tilde{\lambda}(t) = \lambda(T + t)$. Therefore, the computations of its conditional mean and conditional median are relatively straightforward.

Although the mean and median of the predictive distribution are frequently used when predicting the popularity of a tweet, the accuracy of tweet popularity prediction is frequently assessed using the MAPE or MdAPE (Zhao et al., 2015; Kobayashi and Lambiotte, 2016; Mishra et al., 2016). As pointed out by Gneiting (2011), the choice of the point predictor should be consistent with the performance evaluation metric being used to avoid misguided inferences. However, because popularity prediction needs to be made for a large number of tweets where their popularity distributions can be drastically different, the use of squared error or absolute error would be less informative compared to the use of unitless error measures, such as the APE.

Therefore, the MAPE or MdAPE should be used as the evaluation metric when comparing the performances of different popularity predictors. It should be noted, however, that even though the mean and median are optimal predictors when the RMSE and the MAE are used as the respective evaluation metrics, they are, in general, not optimal relative to the MAPE and MdAPE. To be consistent with these two performance evaluation metrics, suitable functionals of the predictive distribution should be used, instead of the mean and median. A discussion on this has been included in Appendix A.

5.4 Application to the Tweet Data

Having presented the form of the intensity assumed by the Poisson process model and the approach employed to obtain the empirical Bayes estimates followed by how these estimates can be used to predict the future popularity of tweets, we shall hereby exhibit the graphical and numerical results obtained alongside the various stages.

5.4.1 Estimated Activity Levels

One of the most important components in the formulation of our model is the rhythm function $\rho(\cdot)$ as shown in (5.1.3) which reflects the diurnal patterns of Twitter users' activity levels. The estimated rhythm function by the KDE with the pseudodata approach for the boundary effect correction is shown in Figure 5.4.1, which suggests that the activity levels are at their peak between 23:00 and 03:00 UTC. On the contrary, the activity levels plummet to their lowest point at around 14:00 UTC.

Besides being useful in gaining insights on the active hours of typical Twitter users, Figure 5.4.1 also indicates the times when a tweet is likely to attract more retweets. For instance, to maximize the potential number of retweets, one should choose to tweet during the peak hours of activities, when many other users are actively engaged with the microblogging platform. This is intuitive in the sense that the densely clustered number of tweets made during these times should be correlated to the attention received by tweets posted by other users, since tweeters can essentially be retweeters themselves, thereby portraying a cyclical pattern of interactivity.

Our implementation relies on the fact that both time zero and time one in Figure 5.4.1 account for 06:00 UTC, following the continuity condition in (5.1.4), which implies that $\hat{\rho}(0) = \hat{\rho}(1) \approx 1.1$, and that both ends in the figure at 0:00 UTC correspond to $\hat{\rho}(0.75) \approx 1.3$. The incorporation of such diurnal patterns is useful in improving the accuracy of tweet popularity predictions, especially at earlier censoring times when the circadian rhythms are in strong effect.

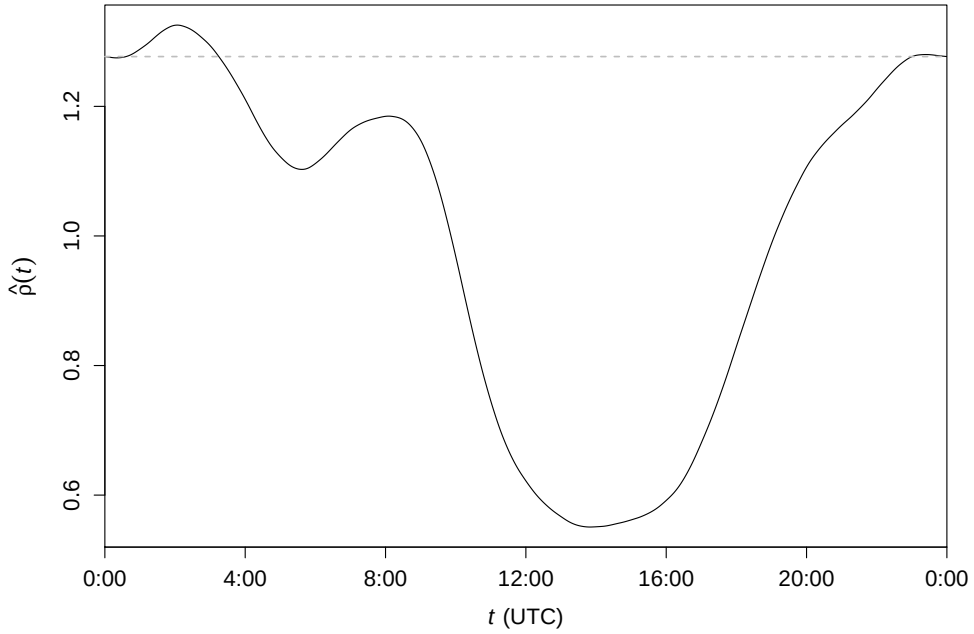

Figure 5.4.1: Estimated function $\rho(\cdot)$ showing the diurnal patterns of Twitter users' activity levels, which suggests that the peak hours of activities are between 23:00 and 03:00 UTC. Conversely, the hours between 12:00 and 16:00 UTC are rather dormant in the activity levels. The horizontal dotted line is used to indicate the continuity of both ends in the plot.

5.4.2 Estimates from the Training Data

Each of the retweet cascades in the training data set was estimated by maximizing over the log-likelihood function in (5.2.2), with the censoring time set to seven days. The logarithms of the ML estimates obtained were then used to construct the prior distribution for model parameters needed when predicting the popularity of a tweet in the test data set. The statistics of the estimated log-parameter values are summarized in Table 5.4.1. As indicated in the table, the median estimates of

Table 5.4.1: The summary statistics of the log-parameters obtained using the ML estimation approach based on the training data set, at $T = 7$ days. The median estimates for $\log \hat{\alpha}$, $\log \hat{\beta}$, and $\log \hat{\gamma}$ are 9.535, 5.617, and 0.407 respectively.

	Min	Q_1	Q_2	Q_3	Max	Mean
$\log \hat{\alpha}$	3.454	8.485	9.535	10.355	42.185	9.297
$\log \hat{\beta}$	−37.892	4.098	5.617	6.629	41.207	4.646
$\log \hat{\gamma}$	−25.200	0.206	0.407	0.687	14.285	0.970

the log-transformed parameters, namely $\log \hat{\alpha}$, $\log \hat{\beta}$, and $\log \hat{\gamma}$ are 9.535, 5.617, and 0.407 respectively.

The summary statistics in Table 5.4.1 are naturally acquired based on $\log \hat{\alpha}_i$, $\log \hat{\beta}_i$, and $\log \hat{\gamma}_i$, for $i = 1, 2, \ldots, 71815$, which, when used in the EB estimation approach, are denoted respectively by $\log \hat{\alpha}_i^0$, $\log \hat{\beta}_i^0$, and $\log \hat{\gamma}_i^0$. As detailed in Sec-

tion 5.2.3, we ought to fit three separate nonparametric regression models based on these estimated log-parameters, with $y_i = \log \hat{\alpha}_i^0$, $\log \hat{\beta}_i^0$, and $\log \hat{\gamma}_i^0$ being the respective response variables, and (m_i^0, t_i^0) being the input variables. The construction of prior distribution based on these estimated log-parameter values instinctively propounds their importance, and therefore their convergence at this stage based on (4.6.1) has to be warranted.

5.4.3 Selecting the Span Parameters

The GCV method was used to select the span parameter values for the nonparametric regression models, or the LOESS regressions, previously discussed in Section 5.2.3. The optimal span values with $\log \hat{\alpha}^0$, $\log \hat{\beta}^0$, and $\log \hat{\gamma}^0$ being the respective response variables are shown in the first row of Table 5.4.2 at 0.003, 0.008, and 0.020 respectively. The suboptimal span values, which will produce nearly identical

Table 5.4.2: Optimal and suboptimal span values for each of the logarithms of the ML estimates. The first row shows the optimal span values, and the subsequent rows show the suboptimal span values in decreasing optimality based on the GCV scores.

$\log \hat{\alpha}^0$		$\log \hat{\beta}^0$		$\log \hat{\gamma}^0$	
Span	Score	Span	Score	Span	Score
0.003	2.0198	0.008	11.9782	0.020	4.5124
0.004	2.0201	0.009	11.9797	0.010	4.5128
0.005	2.0215	0.007	11.9798	0.009	4.5137
0.006	2.0231	0.010	11.9815	0.030	4.5138
0.002	2.0244	0.006	11.9829	0.008	4.5149

model parameters based on the EB estimation approach, and also similar predictions based on various censoring times, have been included in the subsequent rows of Table 5.4.2 for reference.

5.4.4 Empirical Bayes Estimates

The LOESS fitted values of the log-parameters, or equivalently the EB estimates of the log-parameters at censoring time zero, as functions of m^0 and t^0, are illustrated in Figure 5.4.2. From the figure, the span parameters seem to be too small to produce visually smooth regression surfaces. Nonetheless, we can still observe that the magnitude parameter α tends to increase with m^0, the shape parameter γ tends to decrease with m^0, but the dependence of the scale parameter β on m^0, and the dependence of all the parameters on t^0, do not seem to portray any obvious patterns. Although it is possible to obtain visually more pleasing as well as easier-to-interpret

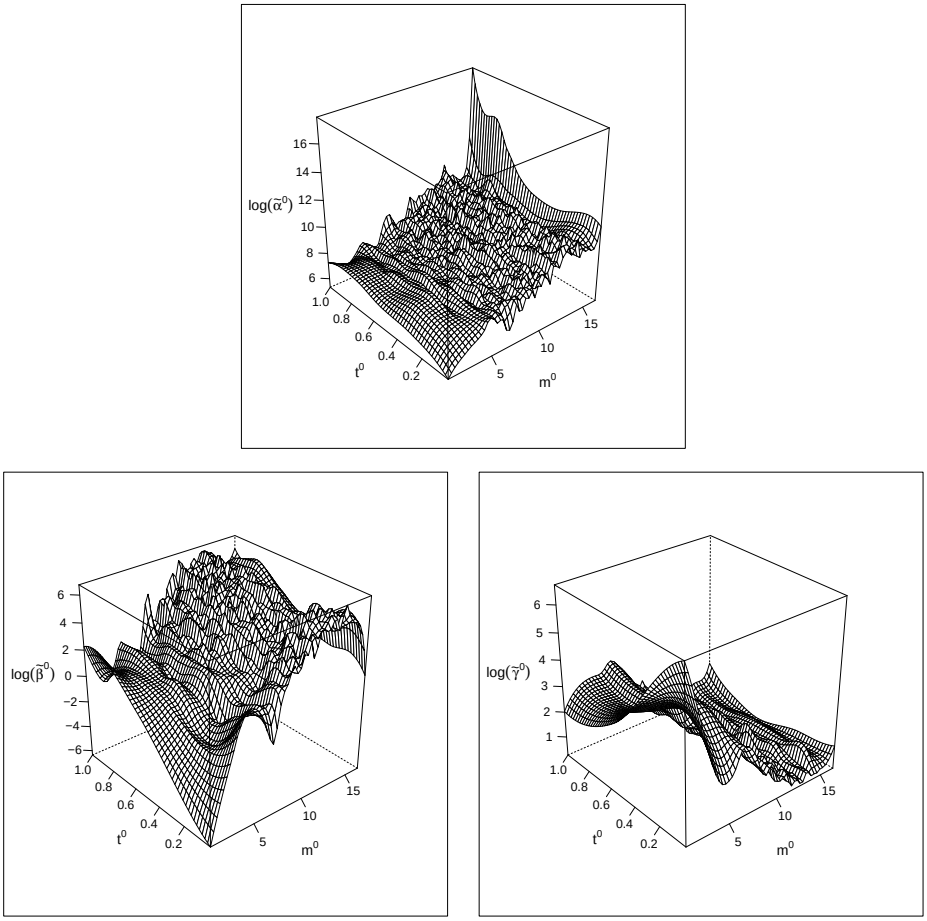

Figure 5.4.2: Logarithms of the EB estimates of the Poisson model parameters as functions of m^0 and t^0 which correspond respectively to the log-transformed number of followers of the original tweeter and the relative publication time of the original tweet. The uneven regression surfaces can be attributed to the small span parameters used.

regression surfaces by using larger span parameters, we have not done so because our primary concern is on prediction rather than estimation.

The EB estimates of the log-parameters, together with their corresponding ML estimates using only the internal history of the retweet cascades, at different censoring times for four randomly selected retweet cascades are illustrated in Figure 5.4.3. It is worth noting that at censoring time $T = 0$, Figure 5.4.3 only shows the EB estimates, but not the ML estimates. This is because the ML estimates are unavailable due to lack of any observations of the corresponding retweet sequences, while the EB estimates are available as the LOESS fitted values based on the training data. Furthermore, the figure also reveals that the EB estimates at different censoring times are substantially more stable compared to the ML estimates, which suggests that the use of the prior distribution has a regularization effect on the ML estimates.

Table 5.4.3 shows the typical log-parameter values found in practice using both the ML and EB estimation approaches, based on the four randomly selected retweet

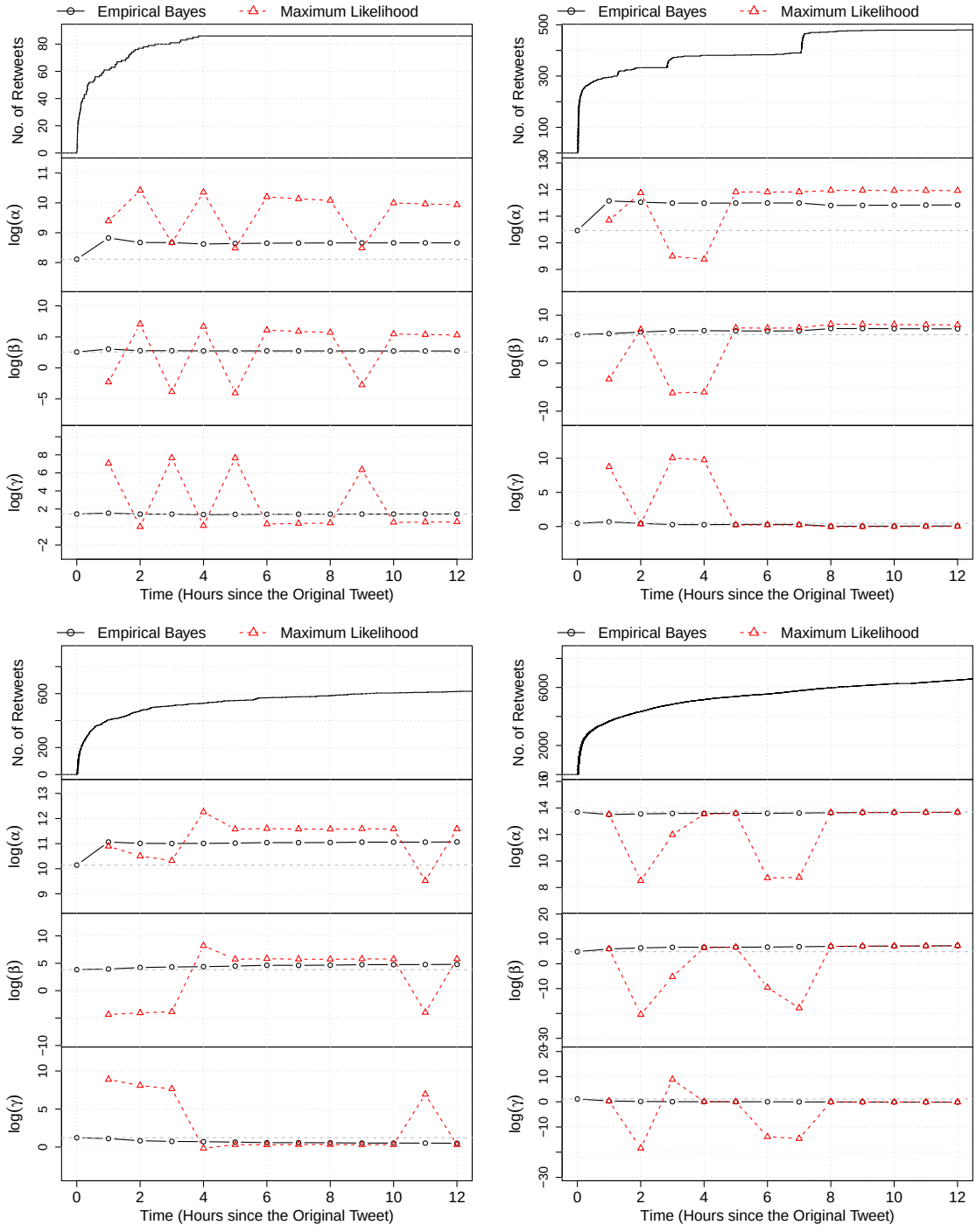

Figure 5.4.3: Estimates of the log-parameters for the Poisson process model using the empirical Bayes (EB) and maximum likelihood (ML) approaches at different censoring times, for four randomly selected retweet cascades. The top panel of each subfigure shows the sample path of the counting process $N(t)$ for the corresponding retweet sequence up to 12 hours, the lower panels of each subfigure show the estimated log-parameters at censoring times $T = 0, 1, \ldots, 12$ hours.

cascades shown in Figure 5.4.3, although we censor them at seven days here instead. We note from Table 5.4.3 that the log-parameters fitted using the ML approach are

Table 5.4.3: The log-parameters for the four randomly sampled retweet cascades in Figure 5.4.3 based on both the ML and EB estimation approaches at $T = 7$ days, together with their respective final popularity values. The degree of changes in the parameter values seems less conspicuous in sample cascade 4, where its final popularity is considerably larger than the other cascades.

Sample	ML			EB			$N(\tilde{T})$
	$\log\hat{\alpha}$	$\log\hat{\beta}$	$\log\hat{\gamma}$	$\log\tilde{\alpha}$	$\log\tilde{\beta}$	$\log\tilde{\gamma}$	
1	9.997	5.488	0.500	8.600	3.066	1.112	93
2	11.872	7.375	0.165	11.443	6.810	0.202	498
3	11.611	6.124	0.148	11.066	5.309	0.221	957
4	13.443	6.392	0.014	13.440	6.384	0.014	9597

denoted by $\log\hat{\theta} = (\log\hat{\alpha}, \log\hat{\beta}, \log\hat{\gamma})$, whilst the log-parameters fitted using the EB approach are denoted by $\log\tilde{\theta} = (\log\tilde{\alpha}, \log\tilde{\beta}, \log\tilde{\gamma})$, which is essentially $\tilde{\eta}$ that appears as the maximizer of the criterion function in (5.2.4).

Interestingly, based on Table 5.4.3 we can observe that the EB estimation approach seems to dampen the estimated log-parameter values for both α and β, which account for the magnitude and scale respectively. On the other hand, the EB estimation approach elevates the estimated log-parameter values for γ which accounts for the shape, except for sample cascade 4 where the value remains unchanged. It is also notable that the difference in the estimated log-parameter values based on the two approaches seems to be the least conspicuous in sample cascade 4, where its final popularity is considerably larger than the other three.

In conjunction with the statistics shown in Table 5.4.1 for the log-parameters obtained via the ML estimation approach, we also include the summary statistics of the log-parameters obtained via the EB estimation approach, at $T = 7$ days and based on the test data set, in Table 5.4.4. It can be observed from the table

Table 5.4.4: The summary statistics of the log-parameters obtained using the EB estimation approach based on the test data set, at $T = 7$ days. The median estimates for $\log\tilde{\alpha}$, $\log\tilde{\beta}$, and $\log\tilde{\gamma}$ are 9.340, 5.205, and 0.461 respectively.

	Min	Q_1	Q_2	Q_3	Max	Mean
$\log\tilde{\alpha}$	4.298	8.682	9.340	9.961	13.766	9.296
$\log\tilde{\beta}$	−5.822	4.298	5.205	5.890	8.984	4.845
$\log\tilde{\gamma}$	−1.384	0.269	0.461	0.722	6.751	0.588

that the median estimates for $\log\tilde{\alpha}$, $\log\tilde{\beta}$, and $\log\tilde{\gamma}$ are 9.340, 5.205, and 0.461 respectively. Compared to the median estimates for the log-parameters obtained via the ML approach, which come at the respective values of 9.535, 5.617, and 0.407,

the magnitude parameter α and the scale parameter β seem to have been deflated, whilst the shape parameter γ seems to have been inflated. This is consistent with the conclusion drawn based on that of Table 5.4.3. The summary statistics of the EB estimates of the log-parameters based on the training data set, through our additional numerical experiments, also reveal very similar values, as exhibited in Table B.2.3.

5.4.5 Prediction Performance Comparisons

Based on the estimated Poisson process model using the empirical Bayes approach at different censoring times, we predicted the final popularity of all the tweets in the test data by the mean, median, median of order -1, and harmonic median of the predictive distribution, and calculated the RMSE, MAE, MAPE, and MdAPE of the predictions. The point predictions at different censoring times using different functionals of the predictive distribution are nearly identical to each other, with the maximum absolute difference between different functionals across the 13 censoring times considered ($T = 0, 1, \ldots, 12$ hours) equals to 9.91.

Table 5.4.5 shows the prediction accuracy of different prediction functionals according to different error metrics at censoring time zero. From this table we can

Table 5.4.5: The prediction accuracy of different prediction functionals at censoring time zero, using the complete test data set. Point predictions based on the predictive mean seem to be consistently more accurate than those based on the other functionals.

	RMSE	MAE	MAPE	MdAPE
Mean	382.47	135.67	47.86%	43.57%
Median	382.57	135.94	48.08%	43.86%
Order (-1) median	382.82	136.23	48.07%	43.96%
Harmonic median	382.86	136.56	48.49%	44.47%

observe that, by any of the four error metrics, the point predictions by the four prediction functionals have comparable accuracy, although the predictions by the predictive mean are slightly yet consistently more accurate than those based on the other functionals. The comparison results at later censoring times are similar, and can be found in Table B.2.2. We have only shown the tables for censoring times $T = 2, 4, \ldots, 12$ hours in the appendix since the patterns portrayed are identical across all the times considered.

It is somewhat surprising that the theoretically optimal functional for a specific error metric does not necessarily lead to a more accurate prediction by the corresponding metric, although in our simulations the optimal functionals do produce slightly more accurate predictions than the other functionals, by the compatible

error metrics. An explanation to this phenomenon is that the numbers of retweets may not follow the Poisson distributions exactly while they do in the simulations. Due to this observation and the ease-of-computation of the mean of the predictive distribution, in the sequel we shall use the predictive mean as the point prediction under the Poisson process model, irrespective of the error metric chosen.

Moreover, based on Table 5.4.5 and Table B.2.2 we can see that the MAE, MAPE and MdAPE all exhibit increasingly better performances with larger censoring times, but the RMSE seems to fluctuate indefinitely. This can be attributed to the presence of grossly erroneous predictions from which their errors are severely magnified by the metric, which then conceal the good performance of the model. Thus, our recommendation regarding the use of unitless error measures such as those based on the APE is further substantiated. Another important thing to note is that the values shown in Table B.2.2 are not directly comparable to those of Table 4.5.3 and Table 4.5.4, since the values in Table 4.5.3 and Table 4.5.4 require the exclusion of a very small amount of outliers. The assessments of prediction accuracy shown in Table 5.4.5 and Table B.2.2, on the other hand, take into consideration all the retweet cascades in the test data set, without excluding any values.

Therefore, to compare the performance of the prediction method proposed in this chapter with those of the competing approaches in the literature, we shall use the prediction APE. The choice of this evaluation metric, as discussed before, is partly due to the highly heterogeneous tweet popularity levels, demonstrated in Table 1.3.3. Furthermore, as our MaSEPTiDE model in Chapter 4 and the TiDeH model of Kobayashi and Lambiotte (2016) have been shown to outperform the other competing models, such as the SEISMIC of Zhao et al. (2015), we shall only compare the prediction performance of our model proposed herein with these two specific models.

Figure 5.4.4 shows the boxplots of APEs based on the final popularity predictions at $\tilde{T} = 7$ days, by the approaches based on the EB Poisson model, the Poisson model, the MaSEPTiDE model, and the TiDeH model, at censoring times $T = 0, 1, \ldots, 12$ hours. Note that the Poisson model labelled in Figure 5.4.4 refers to the model with its parameter estimated based on the ML approach, and the *EB Poisson model* refers to that estimated based on the EB approach. Both these models have been included in Figure 5.4.4 to show how the incorporation of prior knowledge can help improving the overall prediction performance.

Because the distributions of the APEs of the methods shown have very long right tails, the outlying APE values have not been shown in the boxplots for better visualization. Note, at time zero, only the EB Poisson model can produce predictions while the other methods are not able to produce any predictions at all due to the lack of the parameter estimates. Moreover, recall that the smoothing parameter used in

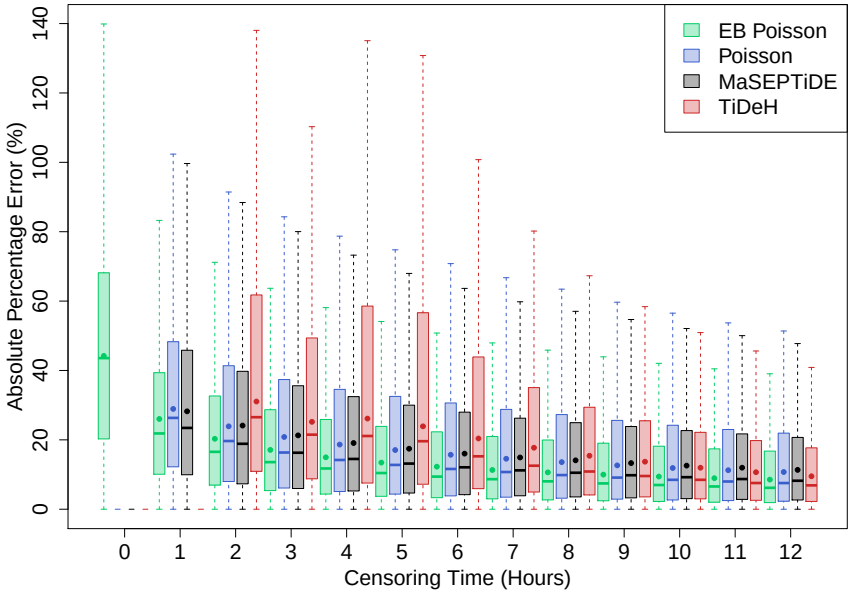

Figure 5.4.4: The APEs of different prediction methods across different censoring times at $T = 0, 1, \ldots, 12$ hours. The Poisson model has been included for the purpose of comparisons with its EB counterpart. The circular point in each boxplot shows the MAPE, while the horizontal thick bar shows the MdAPE. The EB Poisson model is clearly the best performing model at all the censoring times, and is able to make a prediction even at time zero.

the nonparametric estimation for the infectivity function of the TiDeH model is set at one hour. This impedes the approach from producing any meaningful predictions at $T = 1$ hour, and is thus excluded from the comparison at that time.

From Figure 5.4.4 we note that the horizontal thick bar in each boxplot refers to the MdAPE and the circular point refers to the MAPE. For each prediction method under evaluation, both the MdAPE and MAPE decrease as the censoring time increases, indicating a gradual improvement in the prediction accuracy. More importantly, the EB Poisson model seems to consistently outperform the other competing approaches across all the censoring times based on both the metrics.

It should be noted that, for fair comparisons, the Poisson model, the MaSEPTiDE model, and the TiDeH model approaches have also incorporated the knowledge on the lower bound of the predicted final popularity. Nonetheless, even without imposing the lower bound, the EB Poisson model would still stand out as the best performing model, as proven in Figure B.1.2. Thus, it can be concluded that the EB Poisson model serves as an efficient and powerful popularity prediction method.

5.5 Discussion

The assessment of the goodness-of-fit shall be presented here as an additional discussion, since it is useful in unveiling how good our Poisson and EB Poisson models

are, in terms of their capabilities to describe the historical data. We first note that the proposed EB approach is essentially a penalized maximum likelihood approach which poses a larger curvature on the likelihood function, and so its model fit is not expected to be as good as that based on the ML estimation approach. This also implies that a good model fit to the observed data does not necessarily lead to a more accurate popularity prediction.

The goodness-of-fit of the Poisson model and the EB Poisson model can be assessed similarly using the residual point process approach based on Papangelou's random time change theorem, presented in Section 2.6 and used in a similar manner in Section 4.5.2. In essence, to assess the goodness-of-fit of the Poisson model, we can assess the uniformity of the transformed event times $\hat{\Lambda}(\tau_i)$, $i = 1, 2, \ldots, N(T)$, in the interval $(0, \hat{\Lambda}(T)]$ where $\hat{\Lambda}(t) = \Lambda(t; \hat{\theta}) = \int_0^t p(s; \hat{\theta})\hat{d}(s)\,\mathrm{d}s$ using a similar Kolmogorov-Smirnov test of uniformity. The goodness-of-fit of the EB Poisson model can also be assessed similarly, by replacing $\hat{\theta}$ with $\tilde{\theta}$.

The results of the assessments are illustrated in Table 5.5.1, where the upper panel shows the percentages of cascades passing the goodness-of-fit test in the training data based on the ML estimation approach, and the lower panel shows those passing in the test data based on the EB estimation approach. By referring to both

Table 5.5.1: The percentages of cascades where the Poisson model (upper panel) and the EB Poisson model (lower panel) pass the goodness-of-fit test, at different significance levels and censoring times. At significance level of 0.01, the percentages of cascades passing the test using data accumulated in the first 12 hours for the Poisson model and EB Poisson model are 74.6% and 50.4% respectively.

Significance level	Censoring time (hours)						
	2	4	6	8	10	12	168
0.01	77.7%	76.3%	75.6%	75.1%	74.9%	74.6%	69.3%
0.05	71.6%	70.7%	69.9%	69.4%	68.9%	68.3%	61.5%

Significance level	Censoring time (hours)						
	2	4	6	8	10	12	168
0.01	63.4%	58.1%	55.2%	53.2%	51.5%	50.4%	43.3%
0.05	48.6%	43.3%	40.6%	38.7%	37.3%	36.3%	30.4%

panels in the table, the decrease in the goodness-of-fit when the EB estimation approach is used instead of ML estimation approach is quite substantial. Specifically, at the censoring time of 12 hours, the ML estimation approach has successfully fitted around 75% of the retweet cascades in the training data, based on the significance level of 0.01. This value drops to around 50% when the EB estimation approach is used, although the approach was applied on the test data.

To make the interpretation more convincing, we have also included the results of the goodness-of-fit assessment for the EB Poisson model based on the training data

set, in Table B.2.4. We have only shown the assessment results at censoring times $T = 2, 4, \ldots, 12$ hours in this section, since those based on $T = 1, 3, \ldots, 11$ hours have rather consistent changes, and should be self-explanatory. Also, we note that the goodness-of-fit test cannot be run without first observing some retweet events, and so we have excluded the assessment at time zero.

5.6 Concluding Remarks

In this chapter we have proposed a simple Poisson process model for the sequence of retweet times of a tweet and a novel Empirical Bayes (EB) type approach to fit the model. Although the Poisson process model is not expected to provide better fit to the retweet time sequences than the more elaborate models available in the literature, when used with the EB approach for inference, this simple model was found to produce overall more accurate tweet popularity predictions. An additional important advantage of the proposed approach is its ability to produce a prediction at censoring time zero before any retweets occur, or making a pre-publication tweet popularity prediction.

The proposed EB approach of tweet specific parameter estimation is essentially a penalized maximum likelihood approach whereby a concave quadratic penalty is added to the log-likelihood function and penalizes parameters further away from the initial nonparametric regression estimators of the parameters more than those nearer. Because of the presence of the quadratic penalty, the penalized log-likelihood function tends to have larger curvature than the unpenalized log-likelihood, and therefore it is much easier to maximize than the original log-likelihood. Indeed, during our numerical analysis, we were always able to obtain a reasonable MAP estimator, or maximum penalized likelihood estimator, of the tweet specific parameters, and a sensible popularity prediction, on each of the retweet cascades in the test data set. This is commendable compared to the state-of-the-art popularity prediction approaches in the literature which may produce grossly erroneous popularity predictions or fail to produce any predictions at all on a number of tweets in the data set.

As mentioned earlier, the EB approach to incorporate knowledge in the training data external to a specific retweet time sequence when estimating the model parameters, can also be applied on other point process models. In fact, our numerical experiments with the EB approach on the MaSEPTiDE model seem to suggest even better prediction performance than the EB Poisson model when the retweet data has been accumulated and observed for a reasonably long time. Nonetheless, the EB MaSEPTiDE model still requires substantially longer computational time than the EB Poisson model.

In selecting the bandwidth parameter used by the kernel density estimator to estimate the circadian rhythm function in the retweet intensity, we have also experimented with other bandwidth selectors, such as the solve-the-equation and direct plug-in methods of Sheather and Jones (1991). These alternative approaches all tend to produce much smaller bandwidths and similar but more wiggly density curve estimates than the default normal reference approach. The wiggly curve estimates can cause the numerical integration needed in evaluating the log-likelihood function in (5.2.2) and the criterion function in (5.2.4) to converge very slowly or even break down altogether, and when the estimation based on such wiggly curve estimates works out, the resulting popularity prediction does not differ materially from that based on the estimates obtained using the default bandwidth selector. Therefore, we recommend the use of the default bandwidth selector adopted by R when estimating the rhythm function.

An important assumption we make when estimating the circadian rhythm function is that all tweeters and retweeters share the same rhythm function, which might not hold given that the Twitter users are likely to come from different time zones. Therefore a stratification of the tweets according to the time zone from which the original tweeter comes from should be able to further improve the prediction accuracy of our approach. However, this has not been implemented due to the lack of relevant location information of the tweeters. To collect information on the geographical locations of the tweeters, we note that the tweets posted must be geotagged, which first require the geolocation services to be manually enabled by the tweeters.

The construction of the prior distribution for the tweet specific parameters in our empirical Bayes approach has been inspired by the concept of confidence distribution, which, as we have noted above, leads to an interpretation of the maximum a posteriori estimator or the maximum penalized likelihood estimator. A natural question to ask is whether the penalty implied by the choice of the prior distribution is optimal in any sense. To answer this question we have also experimented with other choices of prior distributions for the parameters, such as the predictive distribution for the parameters when they are treated as the response variables in the nonparametric regression step. However, with the other choices of the prior distributions we have experimented, the accuracy of the popularity predictions tends to worsen. Therefore the prior distribution motivated by the confidence distribution seems optimal to some extent. Still, the prior distributions suggested by other regression methods to learn the functional dependence of the tweet specific parameters on the input variables can potentially lead to even more accurate popularity predictions. A systematic study on the choices of the prior distributions in the EB framework might be interesting future work.

Chapter 6

The Empirical Bayes Approach Applied on Alternative Models

The usefulness of the empirical Bayes (EB) type approach in combining the knowledge external to a specific retweet time sequence with that observed internally up to a certain censoring time has been well demonstrated in Chapter 5. A notable feature of the EB approach is its regularization effect on the maximum likelihood (ML) estimates. This is accomplished through the addition of a concave quadratic penalty to the log-likelihood function, which imposes greater penalties on parameters further away from the initial nonparametric regression estimators than those nearer. Therefore, the EB approach is efficacious in eliminating parameters likely to produce erratic or nonsensical prediction values, as suffered by a small amount of retweet cascades predicted via the MaSEPTiDE model approach. An additional advantage of the EB approach, as we have mentioned, is its capability to perform pre-publication tweet popularity prediction.

On another note, by the original data presented in Section 1.3, the only information readily available for any cascade at time zero is n^0 and t^0, which correspond respectively to the number of followers of the tweeter and the relative posting time of the tweet. Nonetheless, the contents of tweets can still be extracted by using the Twitter application programming interface (API). This in turn enables us to analyze the sentiments of the extracted contents based on their semantic features, and allows us to inspect if any further improvement to the existing models is possible. Such an objective can be achieved by, say, including the sentiment values obtained as input variables used in the nonparametric regression step. The practicality of sentiment analysis under this context shall be briefly discussed in this chapter.

For extra clarity on the prediction performances of various models, we shall focus on presenting the numerical results in this chapter. Also, for the purpose of consistency and ease-of-interpretation, these numerical results shall be based on point predictions using the predictive mean, with the MAPE and MdAPE being the

performance evaluation metrics. We have also incorporated the extra knowledge on the lower bound of the final popularity of tweets for all the models considered herein, although similar conclusions can be drawn even without such implementation.

The remainder of this chapter is organized as follows. We first describe the MaSEPTiDE model employing the EB approach in Section 6.1. To further testify the applicability of the EB approach on different point process models, a parametric version of the TiDeH model based on the approach is subsequently presented in Section 6.2. For supplementary purposes, we also present in Section 6.3 a viable alternative to the aforementioned EB Poisson model which makes use of the results obtained from sentiment analysis. Finally, we give some concluding remarks in Section 6.4.

6.1 The Marked Self-Exciting Point Process Model

Despite being able to make accurate tweet popularity predictions based on early censoring times, the MaSEPTiDE model has some noticeable limitations. First, it is unable to make any popularity prediction without accumulating some events beforehand. Second, the estimation of its parameters and the prediction based on these estimated parameter values can be computationally expensive for a number of retweet cascades. Third, the predictions based on both the solve-the-equation and simulation-based approaches can sometimes be unreasonably large. These addressed issues, however, can be alleviated by using the EB approach, as we shall demonstrate in the following nested sections.

6.1.1 Parameter Estimation and Prediction

The conditional intensity assumed by the MaSEPTiDE model, together with its parametric components, have been shown through (4.1.1)-(4.1.3). The model consists of the parameters $\theta = (\alpha, \beta, \gamma, \delta_1, \delta_2)$, which, as we have mentioned, can be obtained for each retweet cascade via the ML estimation approach, through maximizing the likelihood function in (4.2.1) over the interval $[0, T]$. These ML estimates, when obtained at the censoring time of seven days for all the retweet cascades in the training data, can be used to setup the prior distribution for model parameters used in the EB estimation approach, which involves the analogous procedures discussed in Section 5.2.3.

Briefly, the steps include obtaining the ML estimates for all the retweet cascades in the training data set, denoted by $\hat{\theta}_i^0 = (\hat{\alpha}_i^0, \hat{\beta}_i^0, \hat{\gamma}_i^0, \hat{\delta_1}_i^{\,0}, \hat{\delta_2}_i^{\,0}), i = 1, 2, \ldots, 71815$, fitting five separate nonparametric regression models with $y_i = \log \hat{\alpha}_i^0$, $\log \hat{\beta}_i^0$, $\log \hat{\gamma}_i^0$, $\log \hat{\delta_1}_i^{\,0}$, $\log \hat{\delta_2}_i^{\,0}$ as the respective response variables and $x_i = (m_i^0, t_i^0)$ as the input variables, predicting the values of log-parameters $\eta = (\eta_1, \eta_2, \eta_3, \eta_4, \eta_5) = (\log \alpha,$

$\log\beta$, $\log\gamma$, $\log\delta_1$, $\log\delta_2$) based on these fitted regression models for any new input $x = (m^0, t^0)$, and denoting the predicted log-parameters by $\tilde{\eta}^0 = (\tilde{\eta}_1^0, \tilde{\eta}_2^0, \tilde{\eta}_3^0, \tilde{\eta}_4^0, \tilde{\eta}_5^0) = (\log\tilde{\alpha}^0, \log\tilde{\beta}^0, \log\tilde{\gamma}^0, \log\tilde{\delta}_1^0, \log\tilde{\delta}_2^0)$ with the standard errors $(e_1, e_2, e_3, e_4, e_5)$. The prior density for the log-parameters η, following (5.2.3), can then be defined as follows,

$$\pi(\eta) = f(\eta_1; \tilde{\eta}_1^0, e_1^2) f(\eta_2; \tilde{\eta}_2^0, e_2^2) f(\eta_3; \tilde{\eta}_3^0, e_3^2) f(\eta_4; \tilde{\eta}_4^0, e_4^2) f(\eta_5; \tilde{\eta}_5^0, e_5^2),$$

where $f(\cdot; \mu, \sigma^2)$ denotes the normal density function with mean μ and variance σ^2, and the estimate $\tilde{\eta}^0$ should appear as the maximizer of $\pi(\eta)$. The external knowledge based on the log-prior density and the internal knowledge based on the current log-likelihood can then be combined into the following criterion function,

$$\tilde{\ell}(\eta) = \log\pi(\eta) + \ell(e^\eta),$$

where $\ell(\cdot)$ has been given in (2.5.2), with its intensity and component functions defined through (4.1.1)-(4.1.3). Recall that at censoring time T, the estimator for the parameters θ is $\tilde{\theta} = e^{\tilde{\eta}}$, where $\tilde{\eta}$ is the maximizer of $\tilde{\ell}(\eta)$, or equivalently the MAP estimator of η. When the parameters of the MaSEPTiDE model are estimated by this convention, the model can be referred to as the *EB MaSEPTiDE* model.

The prediction procedures are similar to those explained in Section 4.4, except that under the EB framework we can treat m_i as an additional parameter which depends solely on the external knowledge at time zero, with increasingly more weight imposed on the internal knowledge as the censoring time T increases. This value is needed in the expected response $\hat{R}$ when using the solve-the-equation approach, and will be iteratively sampled from when using the simulation-based approach. In fact, we can even use a more crude approach, say, when no retweet event has been observed internally, we choose a cascade in the training data which is of the closest distance based on (m^0, t^0), and use the log-transformed average number of followers in $\hat{R}$ of the solve-the-equation approach or its empirical distribution function $\hat{F}$ in the simulation-based approach, and when an event has arrived, we revert to using the internal knowledge for prediction.

6.1.2 Numerical Results

The numerical results for the procedures described in Section 6.1.1 shall be presented here. First, the span parameter values used in the locally weighted kernel regressions for the logarithms of the ML estimates $\log\hat{\alpha}^0, \log\hat{\beta}^0, \log\hat{\gamma}^0, \log\hat{\delta_1}^0$, and $\log\hat{\delta_2}^0$ on m^0 and t^0 come at the respective values of 0.005, 0.030, 0.060, 0.020, and 0.004. These values were similarly selected based on experimenting with different span parameters

using the GCV approach.

The median estimates of the log-parameters based on the ML approach applied on the training data at the censoring time of seven days, namely $\log\hat{\alpha}$, $\log\hat{\beta}$, $\log\hat{\gamma}$, $\log\hat{\delta_1}$, and $\log\hat{\delta_2}$ have the respective values of 3.935, -2.687, 1.815, 0.350, and -4.994, as shown in left panel of Table 6.1.1. The right panel of Table 6.1.1, in tandem, shows the estimated log-parameter values based on the EB approach, for $\log\tilde{\alpha}$, $\log\tilde{\beta}$, $\log\tilde{\gamma}$, $\log\tilde{\delta_1}$, and $\log\tilde{\delta_2}$, with the median estimates being 4.298, -1.908, 2.127, 0.418, and -5.436 respectively. Note that the EB estimates shown here were obtained based on the test data set, but those obtained based on the training data set have nearly identical values as well, revealed in Table B.2.5.

Table 6.1.1: The quartiles of log-parameters for the MaSEPTiDE model estimated based on the ML and EB approaches, censored at seven days. The median estimates of the log-parameters by the ML approach are 3.935, -2.687, 1.815, 0.350, and -4.994, while those by the EB approach are 4.298, -1.908, 2.127, 0.418, and -5.436 respectively.

	ML			EB		
	Q_1	Q_2	Q_3	Q_1	Q_2	Q_3
$\log\alpha$	2.863	3.935	4.840	4.009	4.298	4.742
$\log\beta$	-3.852	-2.687	-1.524	-2.567	-1.908	-0.685
$\log\gamma$	-0.035	1.815	4.632	0.944	2.127	3.212
$\log\delta_1$	0.156	0.350	0.582	0.276	0.418	0.579
$\log\delta_2$	-6.350	-4.994	-3.845	-6.315	-5.436	-4.737

From Table 6.1.1 we further note that, with the incorporation of prior knowledge, all the parameter values can fluctuate indefinitely, except for the infectivity parameter β which exhibits an overall increase. By the exponential decay form of the function in (4.1.3), this implies that the infectivity should deteriorate faster. To put that into perspective, using the median estimate based on the ML approach, the time needed for the infectivity to drop to 1% of its initial level is $\log(100)/\exp(-2.687) = 67.6$ seconds, while that based on the EB approach only needs $\log(100)/\exp(-1.908) = 31.0$ seconds. This opines that the infectivity is now decaying twice as fast, and as a result the tweet popularity tends to die out more rapidly. This said, the existence of grossly erroneous predictions based on the original MaSEPTiDE model can be attributed primarily to the infectivity that decays at a disproportionately slow rate, which then causes the high explosiveness of the tweet, much like the supercritical regime portrayed in the framework of the SEISMIC (Zhao et al., 2015).

Table 6.1.2 shows the percentages of large prediction APEs at various censoring times $T = 1, 2, \ldots, 12$ hours based on the conditional expectation of the MaSEPTiDE model. We can observe that at censoring time $T = 1$ hour, the percentages of retweet cascades in the test data set with the APE values of at least $10^3\%, 10^4\%$, and

Table 6.1.2: Grossly erroneous predictions by the MaSEPTiDE model, based on the absolute percentage errors (APEs) at various censoring times. At censoring time $T = 1$ hour, the percentages of retweet cascades in the test data set with the APE values of at least $10^3\%, 10^4\%$, and $10^5\%$ are around 1.6%, 0.4%, and 0.1% respectively. Overall, the smaller censoring times are prone to larger errors.

T (hours)	APE (%)		
	$\geqslant 10^3$	$\geqslant 10^4$	$\geqslant 10^5$
1	1.554	0.390	0.099
2	1.301	0.132	0.081
3	1.097	0.080	0.056
4	1.002	0.088	0.058
5	0.880	0.080	0.067
6	0.818	0.079	0.061
7	0.742	0.075	0.064
8	0.638	0.072	0.059
9	0.572	0.073	0.059
10	0.517	0.067	0.057
11	0.393	0.054	0.047
12	0.334	0.056	0.047

$10^5\%$ are approximately 1.6%, 0.4%, and 0.1%. In addition, the table suggests that the smaller censoring times are more susceptible to larger prediction errors. This can be attributed to the limited time given to accumulate sufficient information needed in the estimations of model parameters for the minority of the retweet cascades. The numbers of grossly erroneous predictions, and hence the APEs, naturally diminish over time, as more information is gathered and the dynamics are leveraged.

It is noteworthy that the values in Table 6.1.2 will all be reduced to zeros when the EB approach is used, which signifies that the problem of grossly erroneous popularity predictions has been coped with. This then leads to a noticeable increase in the overall accuracy of prediction, as we shall show hereinafter. On another note, the APEs obtained based on the Poisson model in (5.1.1) using the ML estimation approach at the various censoring times can also be quite large, although the severity is much less than those suffered by the MaSEPTiDE model. These large APE values can be eliminated similarly by using the EB approach, yielding the aforementioned EB Poisson model.

Although the number of extremely mispredicted tweet popularity at each censoring time seems negligible, with barely 0.1% of them having intolerably large APE values ($\geqslant 10^5\%$), its impact on evaluation metrics such as the MAPE should not be underestimated. The prediction MAPEs for the original MaSEPTiDE model without the exclusion of outliers at various censoring times are shown in Table 6.1.3. The MAPEs of the predictions based on the Poisson model and the EB Poisson model have also been included in the table for comparisons. The prediction MAPEs of the

Table 6.1.3: The MAPEs for the MaSEPTiDE model, the EB MaSEPTiDE model, the Poisson model, and the EB Poisson model at various censoring times $T = 0, 1, \ldots, 12$ hours, based on the complete test data set. The EB approach helps to improve the prediction performances drastically, both for the MaSEPTiDE model and the Poisson model. The EB MaSEPTiDE model seems to outperform the EB Poisson model at all the censoring times, except when $T = 0$.

T (hours)	MAPE (%)			
	MaSEPTiDE	EB MaSEPTiDE	Poisson	EB Poisson
0	-	196.2	-	47.9
1	12690.9	24.4	196.3	27.4
2	7767.1	20.8	104.3	23.4
3	2648.2	18.7	76.4	21.0
4	2179.2	17.2	64.9	19.5
5	2542.3	16.1	59.7	18.3
6	1227.9	15.2	54.2	17.4
7	1312.1	14.4	49.8	16.6
8	1587.2	13.8	44.5	15.9
9	1048.3	13.2	40.5	15.4
10	1131.8	12.7	36.1	14.8
11	806.6	12.3	33.5	14.3
12	797.8	11.8	30.5	13.9

MaSEPTiDE model and the Poisson model are considerably large at all the censoring times, as they have been severely magnified by the presence of outlying values. However, the use of the EB approach, as proven in the table, can drastically improve their prediction performances. More importantly, the EB MaSEPTiDE model seems to outperform the EB Poisson model at all the censoring times considered, except at time zero. A similar conclusion can be drawn when the MdAPE is used as the evaluation metric, as indicated in Table 6.1.4. Compared to when the MAPE is used as the metric to assess the prediction performances of models, the values based on the MdAPEs of predictions, by all the four models considered, seem much more orderly. This is to be expected as the predictive median is resilient to the presence of outliers.

Based on both Table 6.1.3 and Table 6.1.4, the EB MaSEPTiDE model consistently performs better starting from censoring time $T = 1$ hour, but is inferior compared to the EB Poisson model at censoring time zero. This implies that the precise time point at which the EB MaSEPTiDE model starts to outperform the EB Poisson model remains unknown, and can be intriguing to investigate. Therefore, we further attempted popularity predictions with several censoring times under one hour on both the models to secure a more conclusive answer.

The prediction MAPEs and MdAPEs based on both the models at the selected censoring times of $T = 1, 2, 3, 4$ minutes and $T = 5, 10, \ldots, 55$ minutes are shown in

Table 6.1.4: The MdAPEs for the MaSEPTiDE model, the EB MaSEPTiDE model, the Poisson model, and the EB Poisson model at various censoring times $T = 0, 1, \ldots, 12$ hours, based on the complete test data set. The EB MaSEPTiDE model outperforms the EB Poisson model at all the censoring times, except when $T = 0$.

T (hours)	MdAPE (%)			
	MaSEPTiDE	EB MaSEPTiDE	Poisson	EB Poisson
0	-	62.0	-	43.6
1	23.7	18.3	26.3	21.8
2	18.8	14.0	19.7	16.5
3	16.4	11.7	16.3	13.6
4	14.7	10.3	14.2	11.7
5	13.1	9.3	12.8	10.4
6	12.1	8.5	11.6	9.4
7	11.2	7.7	10.7	8.7
8	10.4	7.3	9.8	8.0
9	9.7	6.8	9.1	7.4
10	9.2	6.4	8.5	7.0
11	8.7	6.0	8.0	6.6
12	8.2	5.8	7.5	6.2

Table 6.1.5. By both the metrics, the EB MaSEPTiDE model is a better approach

Table 6.1.5: The MAPEs and MdAPEs of predictions based on the EB MaSEPTiDE model and the EB Poisson model, at censoring times $T = 1, 2, 3, 4$ minutes and $T = 5, 10, \ldots, 55$ minutes. By both the MAPE and MdAPE, the EB MaSEPTiDE model performs better than the EB Poisson model starting from $T = 3$ minutes.

T (minutes)	MAPE (%)		MdAPE (%)	
	EB MaSEPTiDE	EB Poisson	EB MaSEPTiDE	EB Poisson
1	66.4	44.6	42.5	37.7
2	55.4	50.6	39.6	37.9
3	47.7	48.8	36.8	37.3
4	43.4	44.8	34.7	35.3
5	40.4	41.5	32.9	33.4
10	33.5	35.2	27.9	29.6
15	30.8	33.9	25.8	28.9
20	29.4	33.0	24.3	28.3
25	28.5	32.1	23.2	27.4
30	27.7	31.2	22.2	26.5
35	27.0	30.5	21.4	25.6
40	26.4	29.8	20.7	24.6
45	25.8	29.2	19.9	23.9
50	25.3	28.6	19.4	23.2
55	24.8	28.0	18.6	22.4

in predicting the popularity of tweets when $T = 3, 4, \ldots$ minutes. On the contrary, the EB Poisson model is a better prediction method when $T = 1, 2$ minutes, aside

from time zero. Thus, the EB Poisson model remains an efficient and reliable pre-publication tweet popularity prediction method, capable of predicting the popularity of tweets at time zero or slightly beyond time zero with outstanding accuracy.

As a remark, similar to the EB Poisson model, the EB MaSEPTiDE model is not expected to provide a better fit to the retweet cascades than its ML counterpart. The results of the goodness-of-fit assessment for the MaSEPTiDE model based on the ML approach have been previously shown in Table 4.5.2, which, at censoring time $T = 12$ hours and significance level of 0.01 for instance, have roughly 82% of cascades in the training data passing the test. The assessment results based on the EB MaSEPTiDE model using the test data, in contrast, are shown in Table 6.1.6. At the same censoring time and significance level, the passing percentage is only

Table 6.1.6: The percentages of cascades where the EB MaSEPTiDE model passes the goodness-of-fit test, at different significance levels and censoring times, based on the test data. At significance level of 0.01, the percentage of cascades passing the test using data accumulated in the first 12 hours is 60.6%.

Significance level	Censoring time (hours)						
	2	4	6	8	10	12	168
0.01	74.1%	68.8%	65.8%	63.6%	61.8%	60.6%	52.0%
0.05	60.2%	54.7%	51.7%	49.6%	48.0%	46.9%	39.3%

around 61%, which is 21% less than that based on the ML approach. Thus, we note again that although the goodness-of-fit test is useful in determining if a specific model is able to fit the historical data well, it does not necessarily imply a more accurate popularity prediction beyond the censoring time. The assessment results for the EB MaSEPTiDE model applied on the training data, with nearly identical values, have also been appended in Table B.2.6.

6.2 The Time-Dependent Hawkes Model

To demonstrate the applicability of the EB approach on other point process models, we note that the parameters of the TiDeH model in (3.4.2) can be obtained through the ML approach discussed in Section 2.5, instead of the semiparametric approach which relies heavily on the window size parameter shown in (3.4.1) and (3.4.2).

6.2.1 Parameter Estimation and Prediction

Before using the likelihood function in (2.5.2), it is noteworthy that the conditional intensity function assumed by the TiDeH model takes the following specific form,

$$\lambda(t) = p(t) \sum_{i=0}^{N(t-)} n_i \phi(t - \tau_i)$$

$$= p(t) \int_{(0,t)\times\mathcal{N}} n\phi(t-\tau) N(\,\mathrm{d}\tau,\, \mathrm{d}n).$$

By the Fubini's theorem, a change of variables, and the assumed forms of the functions p and ϕ, we have

$$\begin{aligned}
\int_0^T \lambda(t)\,\mathrm{d}t &= \int_0^T p(t) \int_{(0,t)\times\mathcal{N}} n\phi(t-s) N(\,\mathrm{d}s,\, \mathrm{d}n)\,\mathrm{d}t \\
&= \int_{(0,T)\times\mathcal{N}} \int_s^T np(t)\phi(t-s)\,\mathrm{d}t N(\,\mathrm{d}s,\, \mathrm{d}n) \\
&= \int_{(0,T)\times\mathcal{N}} n \int_0^{T-s} p(s+u)\phi(u)\,\mathrm{d}u N(\,\mathrm{d}s,\, \mathrm{d}n) \\
&= \int_{(0,T)\times\mathcal{N}} nf(s) N(\,\mathrm{d}s,\, \mathrm{d}n) \\
&= \sum_{i=0}^{N(T-)} n_i f(\tau_i),
\end{aligned}$$

where the function $f(\cdot)$ is used to accelerate the parameter estimation process through a computationally less demanding integral of the form,

$$\begin{aligned}
f(s) =& \int_0^{T-s} p(s+u)\phi(u)\,\mathrm{d}u \\
=& \int_0^{T-s} \alpha_0 \exp\left(-\frac{s+u}{\beta_0}\right) \left\{1 - \gamma_0 \sin\left(\frac{2\pi}{T_d}(s+u+\delta_0)\right)\right\} c \left\{1 \wedge \left(\frac{u}{s_0}\right)^{-(1+\beta)}\right\} \mathrm{d}u,
\end{aligned}$$

with T_d, s_0, c, and β fixed at their respective values presented in Section 3.3 and Section 3.4. Thus, the parameters to be estimated for each cascade in the training data to be used as prior information in the EB approach are α_0, β_0, γ_0, and δ_0.

The implementation of the EB estimation approach based on the TiDeH model follows the same convention in Section 5.2.3, except that the logarithmic transformations for the model parameters have not been implemented due to the presence of negative values in $\hat{\gamma}_0$ and $\hat{\delta}_0$. Therefore, we used the parameters in their original scales to obtain the EB estimates, and predicted the future popularity of tweets using the solve-the-equation or simulation-based approach discussed in Section 3.4. Following the previous convention of nomenclature, the model shall be referred to as the *EB TiDeH* model.

6.2.2 Numerical Results

The results of predictions based on the EB TiDeH model when the MAPE and MdAPE are used as the evaluation metrics are shown in Table 6.2.1. Similar to

Table 6.2.1: The MAPEs and MdAPEs of predictions based on the EB TiDeH model, at censoring times $T = 0, 1, \ldots, 12$ hours. The model seems to have predicted the final popularity of tweets reasonably well, especially from $T = 1$ hour when some retweet events have accumulated.

T (hours)	MAPE (%)	MdAPE (%)
0	354.5	52.9
1	26.2	20.7
2	22.1	16.5
3	19.9	14.4
4	18.3	12.9
5	17.1	11.8
6	16.0	10.9
7	15.2	10.0
8	14.4	9.3
9	13.8	8.6
10	13.2	8.0
11	12.6	7.5
12	12.2	7.0

the EB MaSEPTiDE model, Table 6.2.1 shows that the prediction performance of the EB TiDeH model is inferior compared to that of the EB Poisson model at censoring time zero. Furthermore, although the EB TiDeH model is performing quite decently at later censoring times $T = 1, 2, \ldots, 12$ hours based on both the assessment metrics, the EB MaSEPTiDE model is still relatively more reliable. In contrast, if we based solely on the MAPE values, the EB TiDeH model seems to outperform the EB Poisson model at censoring times $T = 1, 2, \ldots, 12$ hours, but by the MdAPE values, the EB Poisson model appears to outperform the EB TiDeH model at $T = 3, 4, \ldots, 12$ hours. This asserts that both models have comparable prediction performances, except at time zero when the EB Poisson model is clearly the winner. As the EB TiDeH model is considerably better than the original TiDeH model without the incorporation of prior knowledge, the utility of the EB estimation approach is further substantiated.

6.3 The Poisson Model Variant

For all the regression models with different response variables, we have used (m_i^0, t_i^0), $i = 1, 2, \ldots, 71815$ throughout as the input variables, as they are the only information readily available at time zero. That said, whether or not the addition of new input variables such as those based on the semantic features of tweets can improve the prediction performance of the model of interest remains questionable. Specifically, the contents of tweets can be extracted using the application programming

interface (API) based on their respective tweet identification numbers[1], where features of the extracted contents can be used as the input variables to potentially improve the model prediction accuracy.

It is worth mentioning here that the maximum number of tweets that can be fetched from the server is limited to merely 180 tweets per quarter-hour. The extraction process can be conveniently initiated in R software environment for statistical computing (R Core Team, 2016) through the `twitteR` client, together with several other dependencies and complementary packages. Also, access tokens in the forms of a consumer key and a consumer secret are required to setup the open-standard authorization credentials, which allow us to interact with the Twitter server API. These tokens can be generated from an existing Twitter developer applications account.

6.3.1 Retrieving and Using the Sentiment Values

The feature we shall use herein is the tweet sentiment, which, as we have discussed in Section 1.2.1, reflects the writer's attitudes or perceptions towards a specific subject. The sentiment of a tweet in general can be positive, neutral, or negative. For instance, a tweet containing words expressing happiness, kindness, or enthusiasm, asserts a positive sentiment, while that expressing sadness, hatred, or discrimination intuitively asserts a negative sentiment. In contrast, when no emotions can be detected, the tweet sentiment is said to be neutral. It is important to note here that sentiments are inherently subjective, as individuals of varying morals, values, and beliefs may interpret the attitude of the same tweet content rather differently. This calls for the use of different dictionaries in analyzing the tweet sentiments from different contexts.

Getting an artificial intelligence or a computer algorithm to classify tweet sentiments the way humans are capable of doing has been a major challenge. Natural language processing tools aiming to do just that have been abundantly proposed over the recent years, one of which that is relatively popular is the `coreNLP` package of Manning et al. (2014). Performing sentiment analysis based on the package would yield integer-valued outputs which can imply different tweet sentiments, from being very positive to being very negative. A more recent `SentimentAnalysis` package proposed by Feuerriegel and Proellochs (2018) performs similar analysis based on different dictionaries, and returns continuous-valued outputs in the range of $[-1, 1]$. Both these tools have multilingual support, meaning that they can perform sentiment analyses for textual contents of different languages. However, for simplicity we

[1]The status of a tweet can be checked using https://twitter.com/anyuser/status/x, where x is the 18-digit identification number of the tweet

shall focus on tweets made entirely in English, and use Feuerriegel and Proellochs's package for its utility in obtaining more accurate tweet popularity predictions.

Identifying if a tweet has been posted entirely in English is easy, and necessitates a repository containing a comprehensive list of contemporary English words. By filtering out noises like punctuations and numbers followed by splitting the extracted textual content into chunks of words, we can compare the words with those available in the repository, and based on the number of matching words, we can calculate the percentage of English words found in a single tweet, and intuitively if the tweet of interest has been posted entirely in English. However, such practice limits the use of non-standard words like slangs and dialects commonly found in many tweets, and completely disregards tweets made in other languages or consisting mainly of non-textual contents like graphics, emoticons, and symbols.

We have attempted to extract all the tweet contents in the complete data set described in Section 1.3, but some of them are no longer available by the time of extraction. This can be attributed primarily to removal of the contents by the tweeters themselves, or that the contents have been flagged inappropriate and therefore being removed by the administrators. Nonetheless, we still have sufficiently many tweets with their contents remain intact ($\approx 80\%$), although they consist of several different languages. From this pool of available tweets, we randomly chose 10,000 English tweets for further analysis, where half of these tweets were used as the training set, and the remaining ones were used as the test set. We then used the Harvard-IV General Inquirer, Henry's Finance-Specific, Loughran-McDonald, and Quantitative Discourse Analysis Package dictionaries with the respective acronyms of GI, HE, LM, and QD to get the sentiment values for each of the tweets with textual contents.

Figure 6.3.1 shows three sample tweets with their contents, and how the dictionaries perceive their sentiments. Recall that the sentiment values evaluated by the aforementioned dictionaries can come in any value between -1 to 1, where -1 refers to a very negative tweet sentiment and 1 refers to a very positive one. In fact, if we based purely on intuition, we can easily tell that the tweet on the upper panel of Figure 6.3.1 portrays a negative sentiment, while those in the middle and lower panels are of neutral and positive sentiments respectively. It can be seen that the sentiment values returned by the LM dictionary seem to be quite convincing. The values based on the GI dictionary are also reasonable, but those based on the HE and QD dictionaries, especially on the tweet in the upper panel, seem to deviate slightly from the perceived value.

The implementation of the EB approach using the Poisson model has been described in Section 5.2.3, and conveniently summarized in Figure 5.2.1. As our model formulation here is largely similar, we shall only highlight the differences compared

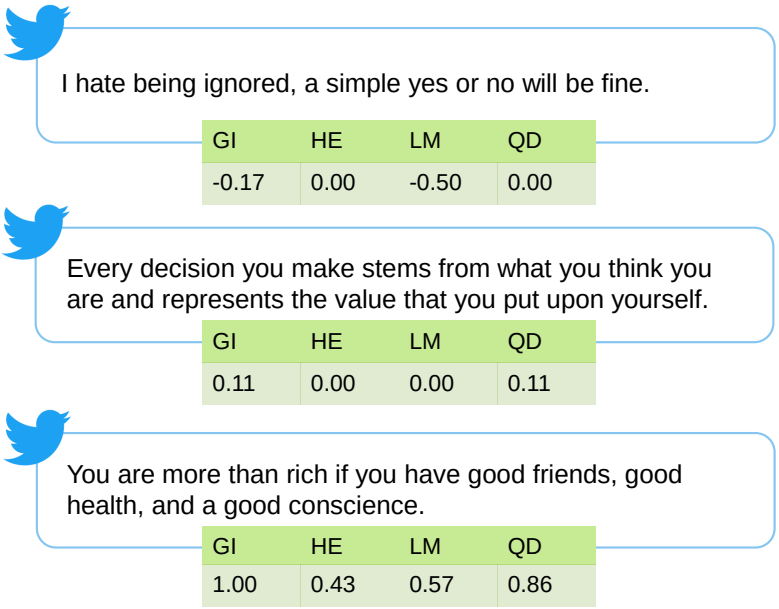

Figure 6.3.1: Sample tweets with their contents evaluated using different dictionaries. The respective sentiment values are shown underneath each tweet with GI, HE, LM, and QD corresponding to the Harvard-IV General Inquirer, Henry's Finance-Specific, Loughran-McDonald, and Quantitative Discourse Analysis Package dictionaries.

to the procedures depicted in Figure 5.2.1. First, we use a slightly different intensity function, where the function $d(\cdot)$ reflecting the circadian rhythm of all Twitter users is removed so that $\lambda(t) = \alpha(1+\beta t)^{-\gamma}$. Second, the input variables used in the nonparametric regression models are now $x_i = (m_i^0, s_i^0)$, for $i = 1, 2, ..., 5000$, with s_i^0 denoting the sentiment value returned by a specific dictionary, and the size i is based on the 5,000 English tweets available in the training data set. Third, as the method is only applicable to tweets with sentiment values, instead of making predictions using the full test data set, we only use its sample of 5,000 English tweets. Tweet popularity predictions at various censoring times can then be performed using similar procedures discussed in Section 5.3.

6.3.2 Numerical Results

Suppose we denote the sentiment values returned by the respective dictionaries for $i = 1, 2, \ldots, 5000$ as $s_{GI_i}^0$, $s_{HE_i}^0$, $s_{LM_i}^0$, and $s_{QD_i}^0$. Table 6.3.1 shows the prediction MAPEs and MdAPEs at censoring times $T = 0, 1, \ldots, 12$ hours when the information on (m^0, t^0), (m^0, s_{GI}^0), (m^0, s_{HE}^0), (m^0, s_{LM}^0), and (m^0, s_{QD}^0) for each retweet cascade in the test data set is used. Note that for (m^0, t^0) which refers to the original EB Poisson model, the prior distribution for its model parameters was built based on the same 5,000 training data, and its predictions were also made using the same 5,000 test data as those used by models based on the different dictionaries, for consis-

Table 6.3.1: The MAPEs and MdAPEs of predictions when the information on (m^0, t^0), (m^0, s^0_{GI}), (m^0, s^0_{HE}), (m^0, s^0_{LM}), and (m^0, s^0_{QD}) for each retweet cascade in the test data set is used. A slight improvement for the model built based on the sentiment values returned by the LM dictionary compared to the original EB Poisson model can be observed from both the MAPE and MdAPE values, starting from $T = 1$ hour and $T = 9$ hours respectively.

T (hours)	MAPE (%)					MdAPE (%)				
	t^0	s^0_{GI}	s^0_{HE}	s^0_{LM}	s^0_{QD}	t^0	s^0_{GI}	s^0_{HE}	s^0_{LM}	s^0_{QD}
0	50.4	51.8	52.5	51.6	52.1	45.5	45.4	45.2	45.9	45.8
1	28.6	29.6	30.0	28.5	29.4	23.5	24.7	24.4	23.8	24.5
2	25.1	25.8	26.2	25.1	25.4	18.8	19.2	20.1	19.7	19.0
3	23.0	23.5	23.6	22.8	22.9	16.2	16.4	17.1	16.8	16.4
4	22.5	23.4	22.4	21.9	22.6	14.3	14.4	15.2	14.9	14.2
5	21.1	21.9	20.9	20.4	21.2	12.9	12.8	13.5	13.4	12.7
6	19.7	20.5	19.5	19.0	19.9	11.7	11.6	12.2	12.0	11.6
7	18.8	19.6	18.4	18.0	18.9	10.7	10.8	11.0	10.9	10.8
8	18.1	18.8	17.6	17.2	18.1	10.2	10.2	10.1	10.2	10.2
9	17.5	18.1	16.9	16.5	17.6	9.6	9.6	9.4	9.4	9.5
10	16.8	17.5	16.3	15.8	17.0	9.0	9.0	8.7	8.8	9.0
11	16.2	16.9	15.6	15.2	16.4	8.5	8.6	8.2	8.2	8.3
12	15.7	16.4	15.1	14.7	15.9	8.0	8.1	7.7	7.8	7.9

tency and comparability. From Table 6.3.1 we note that, by using the MAPE as the evaluation metric, the model constructed based on the sentiment values returned by the LM dictionary seems to slightly outperform the conventional EB Poisson model discussed in Chapter 5, starting from $T = 1$ hour, but by using the MdAPE metric, it only starts to outperforms the EB Poisson model from $T = 9$ hours. Given the overall slight improvement, the proposed EB Poisson model variant utilizing values obtained from sentiment analysis appears to be less practical.

6.4 Concluding Remarks

This chapter revolves around presenting the extended models based on the EB approach, which applies prior knowledge to the estimation of model parameters to prevent the parameters from deviating too much from the typical values. The approach is applicable on various point process models which make use of parametric approaches to estimate their parameters, with indicative examples being the MaSEPTiDE model, the TiDeH model, and the sentiment-based Poisson model.

The first model we demonstrate to have benefited from the EB estimation approach is the MaSEPTiDE model, or under the EB framework we refer it to as the EB MaSEPTiDE model. Despite its commendable prediction accuracy at earlier censoring times, the MaSEPTiDE model has some limitations, notably in terms of the occasionally expensive parameter estimation process and unreasonably large

predicted final popularity values. These problems can be circumvented through the use of the EB estimation approach, which proves that the EB MaSEPTiDE model is a relative faster model capable of producing reliable popularity predictions based on information accumulated in a matter of minutes. Along similar lines, the EB TiDeH model seems to be substantially more stable than its semiparametric counterpart, both in terms of its parameter estimates and its ability to produce more accurate tweet popularity predictions.

As it is rather compelling to testify if the use of sentiment values in point process models can be helpful in improving their prediction performances, we have improvised them in our proposed EB Poisson model. Specifically, we have used integer-valued sentiments in our initial attempts to stratify the nonparametric regression models followed by performing parameter estimations and predictions according to these classes of models, but the results obtained are inferior compared to those of the original EB Poisson model. We have also tried adding in the continuous sentiment values on top of the input variables (m_i^0, t_i^0), but despite the increased complexity, such practice does not seem to produce better results than the model based solely on (m_i^0, s_i^0). Nevertheless, substituting s^0 for t^0 in the EB Poisson model appears to have inconspicuous impact on the outcomes of tweet popularity predictions, asserting that the approach has limited practicality.

As a remark, we note that by using the extracted contents from the API, we can also gain access to additional information such as the character lengths of tweets, which can prove to be useful in some feature-based models. The information on the exact times and dates of tweets and retweets, or better still, with the locations of the tweeters and retweeters available from geotagging services, can also be included in models constructed based on the EB approach to potentially improve their prediction performances. Combining information internal and external to a specific retweet time sequence seems to be useful in producing more reliable popularity forecasts, where the response and input variables may be modified accordingly to adapt to the various needs of different models.

Chapter 7

Conclusion

Numerous popularity prediction methods which can be categorized according the granularity of information used have been proposed over the recent years. Our focus is on local domain prediction under the microscopic level, which translates to predicting the popularity of a tweet based on information available from within the Twitter network at an individual user's level. Such refined attention to single entities can facilitate the identification of influencers to quickly spread information when the needs arise. Examples of prediction methods under this level include the SEISMIC and the TiDeH model, both of which specify the same intensity but differ in the forms of the infectivity functions assumed. Specifically, the SEISMIC uses a nonparametric filtering function to discard posts as they get stale, while the TiDeH model uses a semiparametric circadian rhythm function to exemplify the repetitiveness of human routine activities.

Motivated by the empirical evidence that retweet activities tend to occur in clusters or bursts, we first proposed a marked self-exciting point process model, termed the MaSEPTiDE, to leverage the retweeting dynamics and predict the future popularity of tweets. The memory kernel which describes how the excitation effect due to the original tweet or a retweet is temporally distributed plays a pivotal role here, and is thus incorporated into both the baseline intensity and excitation functions. It is reinforced by two other component functions to contribute to the total excitation effect, one of which reflects how the infectivity of a retweet varies over time, the other pronounces the impact attributable to the number of followers of the retweeter. Based on the form of the intensity beyond the censoring time, the future popularity of a tweet can be predicted, using either a solve-the-equation or a simulation-based approach. The MaSEPTiDE model was found to be capable of accurately predicting the final popularity of tweets, or the total numbers of retweets seven days after their publications, based on substantially less observations than those required by the competing approaches in the literature.

Despite the commendable prediction performance of the MaSEPTiDE model,

it is a post-publication prediction method which still requires the accumulation of retweet events for a considerable amount of time before a prediction can be made. The transient nature of tweets implies that most tweets would reach the peaks of their attention very soon after their publications, and that any meaningful prediction should be made within a matter of minutes. Thus, we proposed an empirical Bayes (EB) type approach to estimate the finite-dimensional parameters of different models through combining internal knowledge on the times of historical retweets up to a certain censoring time, and external knowledge on complete retweet time sequences in the training data. This requires the likelihood to be calculated based on the internal knowledge, and the prior distribution for model parameters constructed based on the external knowledge.

The EB estimation approach is essentially a penalized maximum likelihood (ML) approach which adds a concave quadratic penalty to the log-likelihood function so that parameters further away from the initial nonparametric regression estimators are imposed with heavier penalties than those nearer. Therefore, the penalized log-likelihood function has a larger curvature, and is easier to maximize than its unpenalized version. The prior distribution for the tweet specific parameters under the EB approach has been inspired by the concept of confidence distribution, which leads to an interpretation of the maximum a posteriori (MAP) estimator, or maximum penalized likelihood estimator. Such treatment gives a regularization effect on the ML estimates, and enables pre-publication tweet popularity prediction, or prediction at time zero.

We first employed the EB estimation approach on a relatively simple model where the retweet time sequence is modelled using an inhomogeneous Poisson process, with its intensity function depending on the age of the original tweet and the calendar time, termed the EB Poisson model. The model only requires linear time in the number of retweets to evaluate its likelihood, and the calculation of the future tweet popularity is also rather straightforward. Its remarkable accuracy in predicting the final popularity of tweets is exhibited when contrasted with that of the original Poisson process model, and further borne out when compared to predictions based on the MaSEPTiDE model and the TiDeH model.

To see how other models can benefit from the EB estimation approach, we subsequently applied the approach on the MaSEPTiDE model and the TiDeH model. Using the MaSEPTiDE model, we illustrated how the EB approach can eliminate grossly erroneous predictions occasionally generated by the ML estimates. Backed by the knowledge from the training data, the EB MaSEPTiDE model predicts even more accurately than the original MaSEPTiDE model, although it still requires some observations to outperform the EB Poisson model. The numerical results obtained based on experimenting with the TiDeH model, on the other hand, suggest

that although the EB TiDeH model underperforms the EB MaSEPTiDE model in terms of the prediction accuracy at the censoring times considered, it is comparable to the EB Poisson model, except at time zero. A variant of the EB Poisson model employing the results obtained from sentiment analysis had also been presented, but its practicality was found to be rather limited.

It is worth noting that although the assessment of the goodness-of-fit can provide insights on how well a model fits the historical data, a model with a better goodness-of-fit does not necessarily have a better prediction performance beyond the observation time. This is especially true for EB models which generally predict better than their ML counterparts but have noticeably lower percentages of cascades passing the goodness-of-fit tests. As for the debatable episode on the suitable evaluation metrics to use in assessing the prediction performances of different tweet popularity prediction methods, the MAPE and MdAPE seem appropriate as the tweet data we consider contains highly heterogeneous popularity levels. Although the consistent prediction functionals based on these error metrics are the order -1 median and harmonic median respectively, we have resorted to using the predictive mean as the prediction functional, both for its ease-of-acquisition and the finding that predictions based on different functionals often do not differ materially by the different metrics used.

Although we have attempted to construct models which are both efficient and accurate in predicting the final popularity of tweets to the best of our abilities, there might still be room for further improvements. For example, the circadian rhythm function used in the EB Poisson model can be stratified based on the time zones of the locations wherein the tweeters reside to potentially improve the model prediction accuracy. More generally, we have assumed that the prior distributions of tweet specific parameters used in the EB approach do not have any form of dependency, although learning the functional dependence through other regression methods can possibly lead to even more accurate popularity predictions by various models employing the EB approach.

On another remark, we have used the MAP estimator, or the mode of the posterior distribution, to find the most probable parameter value based on augmenting the optimization objective with prior information, thereby avoiding the more computationally demanding procedures such as the Markov Chain Monte Carlo (MCMC) methods. In fact, even by using the posterior mean as a point estimate of the parameter, our experiments indicate that the prediction performance does not appear to improve at all, and so the choice on using the posterior mode seems well justified.

As supplementary information when making tweet popularity predictions based on the different models proposed in this thesis, we have appended the details of implementation, such as the statistical packages required, the estimated computa-

tional cost of each procedure, the suggested number of simulation replications, and the empirical contribution of each component function in each of the models to the resulting prediction accuracy in Appendix C. Besides being useful in reproducing our results, such information facilitates the constructions of future models with different component functions.

Overall, the EB Poisson model serves as a simple yet powerful prediction tool capable of accurately predicting the final popularity of tweets based on information observed at time zero or slightly beyond time zero. If more observation time is allowed, say, three minutes or longer, then the EB MaSEPTiDE model should be opted for. The ability of the EB models in making accurate popularity predictions based on very short observation times, particularly the EB Poisson model and the EB MaSEPTiDE model, can prove to be useful in various applications, for example in assisting marketing firms and political campaigners to develop effective online advertising strategies on the social networks. Ultimately, as the information diffusion mechanism on Twitter bears a close resemblance to those of other online social networks like Facebook, our models should also be applicable in predicting the popularity of contents found on these platforms.

Appendix A

Optimal Prediction Functionals

The use of the RMSE and MAE in evaluating mean- and median-based predictions have been discussed in the work of Gneiting (2011). However, the MAPE and MdAPE are frequently used in assessing the accuracy and reliability of tweet popularity prediction methods in the literature, which are theoretically inconsistent with the predictive mean and the predictive median respectively. Thus, our discussion here focuses on the optimal functionals for these two evaluation metrics, with special emphasis on how the functionals can be obtained for the models considered in this thesis.

Assume the predictive distribution, F say, is supported by positive reals. Then, as noted by Gneiting (2011), the point prediction that is optimal relative to the MAPE is the *order −1 median* of F, denoted by $\text{med}^{(-1)}(F)$ and defined as the median of the tilted distribution, $(\int_0^\infty y^{-1}\,\mathrm{d}F(y))^{-1}y^{-1}\,\mathrm{d}F(y)$. Here, we note that $\text{med}^{(-1)}(F)$ is defined only when $\int_0^\infty y^{-1}\,\mathrm{d}F(y) < \infty$, which is clearly true in the case we are considering since the predictive distribution has a lower bound of 49, to account for the minimum number of retweets observed over the course of seven days. The point prediction that is optimal relative to the MdAPE, on the other hand, can be shown to be the harmonic mean of the two closest numbers $l \leqslant u$ such that $F(u) - F(l-) \geqslant 1/2$ and $F(u-) - F(l) \leqslant 1/2$, which we refer to as the *harmonic median*, and is conveniently denoted by $\text{hamed}(F)$. It is worth noting here that when F is continuous, the constraints on l and u in the definition of $\text{hamed}(F)$ can be simplified to $F(u) - F(l) = 1/2$.

In general, the computations of the order -1 median and the harmonic median require numerical procedures. A general Monte Carlo approach to compute $\text{med}^{(-1)}(F)$ is to use importance sampling. Specifically, a large i.i.d sample $\mathcal{S} = \{y_i, i = 1, 2, \ldots, B\}$ from the distribution F is simulated first, then a bootstrap resample (with replacement) $\mathcal{S}^* = \{y_i^*, i = 1, 2, \ldots, B\}$ is taken from $\mathcal{S}$ where the selection probabilities for y_i are proportional to y_i^{-1}, and then $\text{med}^{(-1)}(F)$ is approximated by the median of $\mathcal{S}^*$.

Under the Poisson process model considered in this chapter, F is a truncated and shifted Poisson distribution. Therefore, to simulate from F, we can first simulate from the truncated Poisson distribution, with the lower bound $\max\{49 - N(T), 0\}$, using either the rejection method or the inversion method, and then add $N(T)$ to the simulated values. The choice of the method here depends on the value of the lower bound relative to the mean of the (untruncated) Poisson distribution. In particular, when the lower bound is smaller than the Poissonian mean, then the rejection method is more efficient. On the contrary, when the Poissonian mean is much smaller than the lower bound, then the inversion method is more efficient.

Under other point process models, such as the MaSEPTiDE model, the truncated distribution for $N(\tilde{T}) - N(T)$ might not have an explicit or otherwise easy-to-compute density or mass function, and therefore the inversion sampling method is not applicable. Under such a circumstance, we can use the rejection method, that is, by simulating values from the untruncated distribution and retaining only the values which meet the condition of being at least $49 - N(T)$. A potential issue with this rejection method, however, is that none of the values simulated from the untruncated distribution meets the retention condition, despite a large number of values have been simulated. When this happens, we can simply approximate the order -1 median of the predictive distribution by the corresponding lower bound.

For a general predictive distribution F, the computation of its harmonic median can be challenging. However, for the truncated and shifted Poisson distribution under the Poisson process model that we are dealing with here, the numerical computation is relatively easy. First, note that when the mode of the Poisson distribution is $49-N(T)$ at max, the probability function of the truncated Poisson distribution is a decreasing function, therefore $l = 49$, $u = N(T)+\text{med}[N(\tilde{T})-N(T)|N(\tilde{T})-N(T) \geqslant 49-N(T)]$, and $\text{hamed}(F) = 2/(l^{-1}+u^{-1})$. In contrast, when the mode of the Poisson distribution is greater than $49 - N(T)$, we can calculate the harmonic median based on the following algorithm, where $f(\cdot)$ denotes the conditional probability mass function of $N(\tilde{T}) - N(T)$ given that it is at least $49 - N(T)$:

1. Set both l and u to the mode of the Poisson distribution, and if there are two modes, set l to the smaller mode and u to the larger one.

2. While $\sum_{i:l<i\leqslant u} f(i) < 1/2$ is true, repeat the following:

   If $l > \max\{49 - N(T), 0\}$ and $f(l-1) > f(u+1)$, set $l \leftarrow l - 1$; otherwise, set $u \leftarrow u + 1$.

3. Set $l \leftarrow l + N(T)$ and $u \leftarrow u + N(T)$.

4. Return $2/(l^{-1} + u^{-1})$ as the harmonic median.

Under models where the probability mass function of $N(\tilde{T}) - N(T)$ is not available but it is relatively easy to simulate from the distribution, we can try to get a sample from the truncated distribution using the rejection method, and then use the empirical mass function in the above algorithm to obtain an estimate of the harmonic median. When a truncated sample is extremely difficult to acquire, we can again approximate the desired harmonic median by the corresponding lower bound.

Appendix B

Figures and Tables

B.1 Supplementary Figures

B.1.1 The MaSEPTiDE Model

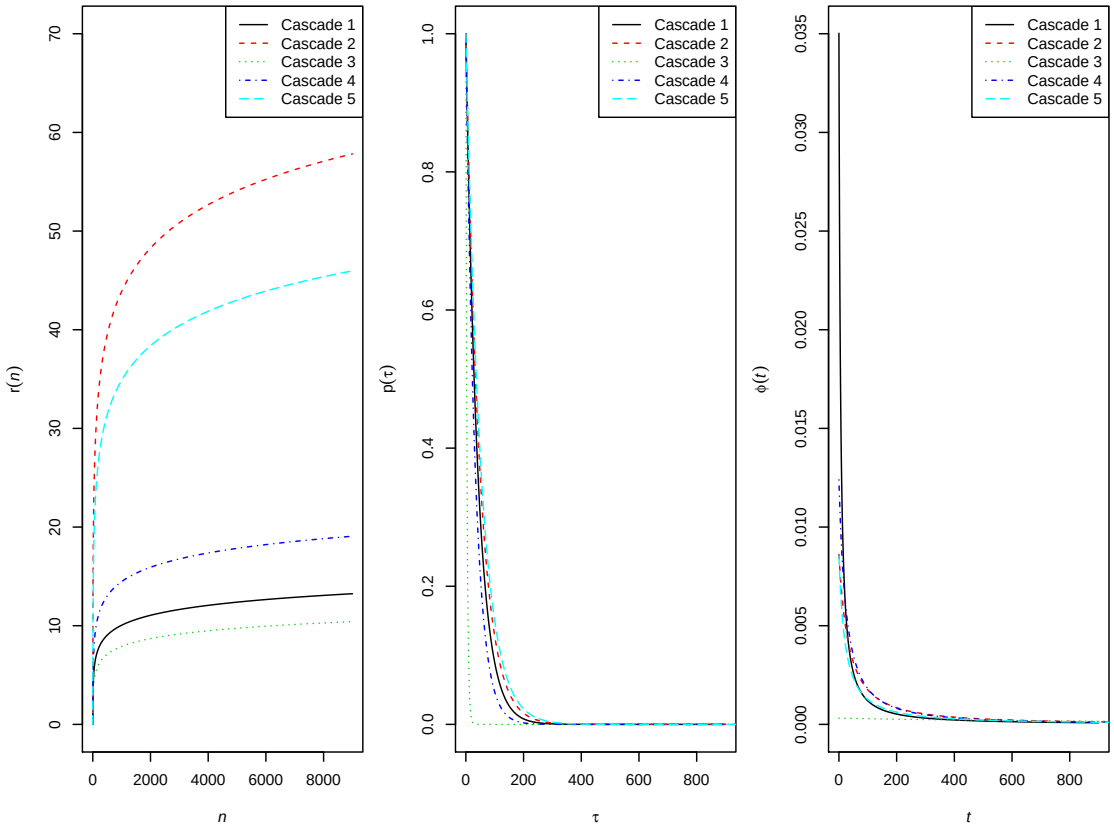

Figure B.1.1: The shapes of the parameters based on the impact function $r(\cdot)$, the infectivity function $p(\cdot)$, and the memory kernel function $\phi(\cdot)$.

B.1.2 The EB Poisson Model

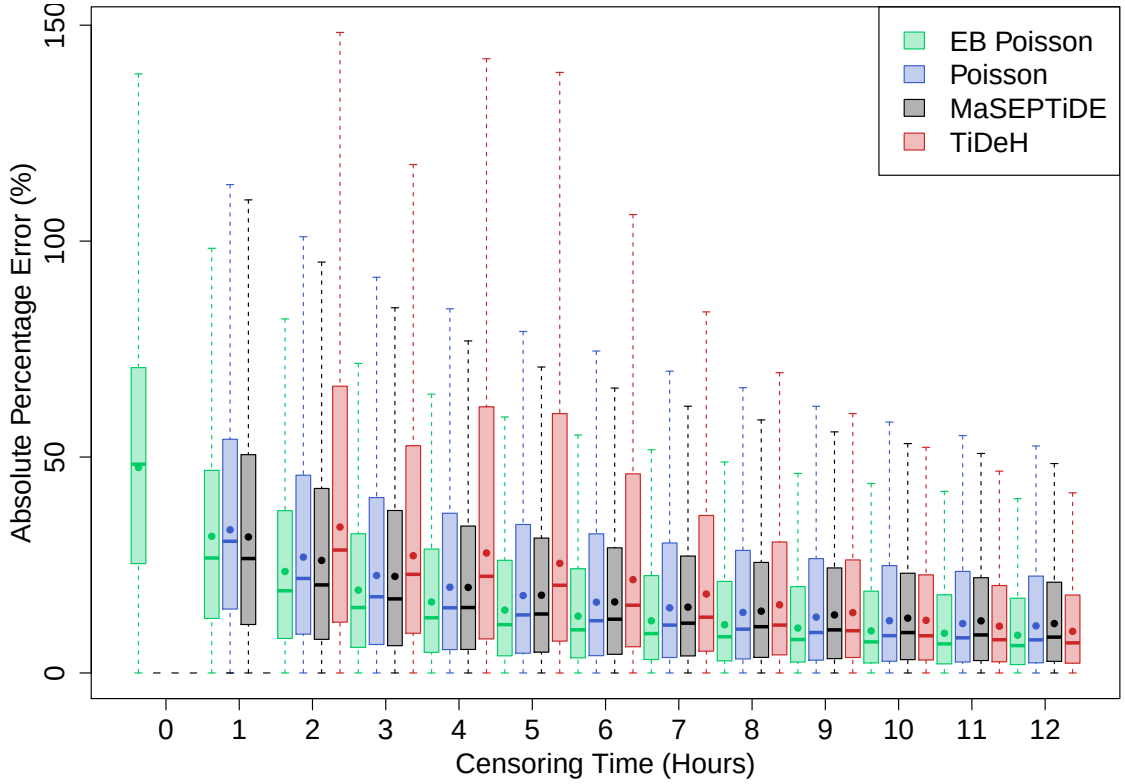

Figure B.1.2: The APEs of different prediction methods across different censoring times at $T = 0, 1, \ldots, 12$ hours, without the adjustments for the lower bounds. The circular point in each boxplot shows the MAPE, while the horizontal thick bar shows the MdAPE. The EB Poisson model is the best performing model at all the censoring times.

B.2 Supplementary Tables

B.2.1 The MaSEPTiDE Model

Table B.2.1: The percentages of retweet cascades with considerably small APE values ($< 5\%$). The retweet cascades are stratified according to the quantile values of popularity. The MaSEPTiDE model consistently outperforms the SEISMIC and the TiDeH model at censoring times $T = 2, 4, 6$ hours except for some very long cascades, where it slightly underperforms the TiDeH model at $T = 6$ hours.

Very short cascades	Censoring time (hours)					
	2	4	6	8	10	12
MaSEPTiDE	23.33	32.53	38.16	43.11	47.42	51.26
SEISMIC	16.56	21.58	25.11	27.51	29.83	31.34
TiDeH	16.27	20.16	27.26	39.70	52.12	59.99

Short cascades	Censoring time (hours)					
	2	4	6	8	10	12
MaSEPTiDE	18.98	26.22	32.00	35.56	39.74	42.87
SEISMIC	14.79	19.07	22.78	25.21	27.33	29.18
TiDeH	14.67	19.27	24.58	34.07	42.29	48.52

Middle-length cascades	Censoring time (hours)					
	2	4	6	8	10	12
MaSEPTiDE	16.61	22.60	27.27	30.90	34.72	37.36
SEISMIC	13.47	16.83	20.88	23.34	24.92	27.73
TiDeH	13.63	18.10	23.02	31.18	37.49	42.89

Long cascades	Censoring time (hours)					
	2	4	6	8	10	12
MaSEPTiDE	16.19	20.56	24.03	26.61	29.83	32.22
SEISMIC	12.42	15.33	18.29	21.37	23.35	25.31
TiDeH	12.48	16.22	20.07	25.36	31.81	36.10

Very long cascades	Censoring time (hours)					
	2	4	6	8	10	12
MaSEPTiDE	13.14	15.03	17.04	19.87	22.01	24.07
SEISMIC	9.95	11.69	13.83	16.09	18.21	20.34
TiDeH	9.15	13.79	18.71	22.24	25.52	29.68

B.2.2 The EB Poisson Model

Table B.2.2: The prediction accuracy of different prediction functionals at censoring times $T = 2, 4, \ldots, 12$ hours, using the complete test data set. Point predictions based on the predictive mean seem to be consistently more accurate than those based on the other functionals.

$T = 2$ hours	RMSE	MAE	MAPE	MdAPE
Mean	390.81	69.73	23.35%	16.54%
Median	390.83	69.86	23.48%	16.73%
Order (-1) median	390.82	69.90	23.51%	16.81%
Harmonic median	390.85	70.03	23.62%	16.95%
$T = 4$ hours	RMSE	MAE	MAPE	MdAPE
Mean	303.99	60.12	19.50%	11.74%
Median	304.01	60.22	19.61%	11.93%
Order (-1) median	303.98	60.24	19.62%	11.94%
Harmonic median	304.01	60.34	19.72%	12.09%
$T = 6$ hours	RMSE	MAE	MAPE	MdAPE
Mean	303.05	55.29	17.37%	9.40%
Median	303.06	55.37	17.46%	9.52%
Order (-1) median	303.02	55.37	17.46%	9.52%
Harmonic median	303.05	55.47	17.55%	9.67%
$T = 8$ hours	RMSE	MAE	MAPE	MdAPE
Mean	349.90	53.12	15.94%	8.03%
Median	349.91	53.19	16.01%	8.12%
Order (-1) median	349.87	53.19	16.01%	8.12%
Harmonic median	349.89	53.27	16.08%	8.23%
$T = 10$ hours	RMSE	MAE	MAPE	MdAPE
Mean	391.89	51.23	14.84%	6.98%
Median	391.89	51.29	14.90%	7.14%
Order (-1) median	391.86	51.28	14.89%	7.14%
Harmonic median	391.88	51.35	14.96%	7.20%
$T = 12$ hours	RMSE	MAE	MAPE	MdAPE
Mean	405.18	49.05	13.86%	6.18%
Median	405.18	49.10	13.91%	6.25%
Order (-1) median	405.14	49.08	13.91%	6.25%
Harmonic median	405.16	49.15	13.96%	6.35%

Table B.2.3: The summary statistics of the log-parameters obtained using the EB estimation approach based on the training data set, at $T = 7$ days. The median estimates for $\log\tilde{\alpha}$, $\log\tilde{\beta}$, and $\log\tilde{\gamma}$ are 9.308, 5.211, and 0.469 respectively.

	Min	Q_1	Q_2	Q_3	Max	Mean
$\log\tilde{\alpha}$	4.851	8.614	9.308	9.949	35.961	9.249
$\log\tilde{\beta}$	−5.637	4.224	5.211	5.898	8.690	4.813
$\log\tilde{\gamma}$	−1.768	0.269	0.469	0.742	6.244	0.601

Table B.2.4: The percentages of cascades where the EB Poisson model passes the goodness-of-fit test, at different significance levels and censoring times, based on the training data. At significance level of 0.01, the percentage of cascades passing the test using data accumulated in the first 12 hours is 50.4%.

Significance level	Censoring time (hours)						
	2	4	6	8	10	12	168
0.01	64.1%	58.4%	55.3%	53.3%	51.7%	50.4%	43.0%
0.05	49.6%	43.9%	41.0%	39.1%	37.6%	36.6%	30.3%

B.2.3 The EB MaSEPTiDE Model

Table B.2.5: The summary statistics of the log-parameters obtained using the EB estimation approach based on the training data set, at $T = 7$ days. The median estimates for $\log\tilde{\alpha}$, $\log\tilde{\beta}$, $\log\tilde{\gamma}$, $\log\tilde{\delta_1}$, and $\log\tilde{\delta_2}$ are 4.303, -1.912, 2.019, 0.424, and -5.404 respectively.

	Min	Q_1	Q_2	Q_3	Max	Mean
$\log\tilde{\alpha}$	-1.144	3.999	4.303	4.774	13.925	4.441
$\log\tilde{\beta}$	-14.207	-2.568	-1.912	-0.658	10.306	-1.503
$\log\tilde{\gamma}$	-7.210	0.834	2.019	3.147	19.303	1.899
$\log\tilde{\delta_1}$	0.001	0.276	0.424	0.589	2.221	0.463
$\log\tilde{\delta_2}$	-14.200	-6.349	-5.404	-4.680	-0.048	-5.646

Table B.2.6: The percentages of cascades where the EB MaSEPTiDE model passes the goodness-of-fit test, at different significance levels and censoring times, based on the training data. At significance level of 0.01, the percentage of cascades passing the test using data accumulated in the first 12 hours is 60.5%.

Significance level	Censoring time (hours)						
	2	4	6	8	10	12	168
0.01	74.5%	68.8%	65.7%	63.6%	61.9%	60.5%	51.9%
0.05	61.1%	55.1%	52.1%	50.2%	48.6%	47.3%	39.5%

Appendix C

Implementation Details

Additional procedural details for the models we have proposed shall be presented here. This includes the statistical packages required, the approximate computation time, the suggested number of simulation replications, and the empirical contribution of individual component function to the resulting prediction accuracy in each of these models.

C.1 Recommended Software and Computation Time

From the estimations of model parameters to the predictions of the final popularity values based on the models proposed in this thesis, we have relied on the R statistical software. The software has numerous built-in functions, but certain packages have to be retrieved from the repository and loaded prior to using them. Specifically, the MaSEPTiDE model and the EB MaSEPTiDE model require the `splines` package for the solve-the-equation approach, the `simPois` function in the `IHSEP` package to simulate inhomogeneous Poisson processes, and the modified `simHawkes1` function in the same `IHSEP` package to generate events based on the cascading algorithm.

For demonstrative purposes, we have selected some random, but somewhat representative retweet cascades, based on the classes of final popularity values in Table B.2.1. The computational costs and the APE values for the proposed models at censoring time $T = 2$ hours are provided in Table C.1.1. Note, the times exhibited in each subtable under Table C.1.1 may differ according to the specifications of the machines used[1], or the resources requested when submitting jobs containing implementation codes to some high performance computational clusters.

Based on the subtables in Table C.1.1 it can be seen that overall, the EB Poisson model is the fastest approach in obtaining the parameter estimates and prediction values. The times required to obtain the parameters for the EB MaSEPTiDE model

[1] we have used a Windows 8.1 computer with core i7-4700MQ processor running at 2.40GHz, topped with 8GB of RAM and 64-bit operating system to obtain the results

are shorter than the original MaSEPTiDE model in general, and on the other hand, the predictions based on the solve-the-equation approach are also substantially faster than those based on the simulation-based approach (at 100 replications) for both the MaSEPTiDE model and the EB MaSEPTiDE model.

If we scrutinize for instance the first sample in the subtable representing very short retweet cascades, it can be observed that it barely costs a second to estimate the parameters for each of the methods proposed. That is, only 0.70 seconds are needed for the MaSEPTiDE model, 0.07 for the EB Poisson model, and 0.64 for the EB MaSEPTiDE model. This translates to the EB Poisson model being roughly ten times faster than the MaSEPTiDE model or the EB MaSEPTiDE model. The predictions based on the solve-the-equation approach for the MaSEPTiDE model and the EB MaSEPTiDE model can also be acquired promptly, requiring only 0.72 and 1.56 seconds respectively. The sole prediction method based on the EB Poisson model yields the prediction result almost instantaneously at 0.10 seconds. Lastly, simulation-based approach prediction by the MaSEPTiDE model and the EB MaSEPTiDE model requires more time, at 34.86 and 38.91 seconds respectively.

Table C.1.1: The computation times required by the key procedures used in the models we have proposed, grouped according to the final popularity values observed and censored at $T = 2$ hours. The results in cells containing two values are obtained from the solve-the-equation and simulation-based approaches respectively. The APE values have also been included for reference.

Very short cascades		Time (seconds)			APE (%)
		Parameter estimation	Prediction	Total	
Sample 1	MaSEPTiDE	0.70	0.72 34.86	1.42 35.56	5.34 4.96
	EB Poisson	0.07	0.10	0.18	4.35
	EB MaSEPTiDE	0.64	1.56 38.91	2.20 39.54	1.82 1.77
Sample 2	MaSEPTiDE	0.43	0.62 8.91	1.06 9.34	16.17 16.09
	EB Poisson	0.09	0.09	0.18	12.79
	EB MaSEPTiDE	0.56	1.63 101.34	2.19 101.90	2.11 2.12
Sample 3	MaSEPTiDE	0.76	0.51 16.70	1.27 17.46	20.93 20.18
	EB Poisson	0.08	0.08	0.16	11.59
	EB MaSEPTiDE	0.66	1.71 36.41	2.36 37.06	1.45 1.35

Short cascades		Time (seconds)			APE (%)
		Parameter estimation	Prediction	Total	
Sample 1	MaSEPTiDE	1.74	0.94 0.96	2.68 2.70	4.87 4.85
	EB Poisson	0.08	0.09	0.18	4.63
	EB MaSEPTiDE	1.19	1.72 49.88	2.91 51.07	0.93 0.39
Sample 2	MaSEPTiDE	1.02	0.56 11.24	1.58 12.25	15.53 15.26
	EB Poisson	0.07	0.08	0.15	15.21
	EB MaSEPTiDE	0.94	1.59 88.71	2.53 89.65	7.77 8.02
Sample 3	MaSEPTiDE	0.78	0.51 17.24	1.28 18.01	22.65 21.86
	EB Poisson	0.09	0.10	0.19	8.21
	EB MaSEPTiDE	0.52	1.50 48.44	2.03 48.96	5.46 3.99

Middle-length cascades		Time (seconds)			APE (%)
		Parameter estimation	Prediction	Total	
Sample 1	MaSEPTiDE	3.70	0.50 3.27	4.20 6.97	5.77 5.89
	EB Poisson	0.09	0.10	0.19	4.11
	EB MaSEPTiDE	3.47	1.65 99.18	5.12 102.65	3.54 3.59
Sample 2	MaSEPTiDE	1.03	0.50 16.77	1.53 17.80	14.56 15.91
	EB Poisson	0.07	0.09	0.16	14.35
	EB MaSEPTiDE	0.78	1.76 39.62	2.54 40.41	11.28 10.97
Sample 3	MaSEPTiDE	2.47	0.58 18.53	3.04 21.00	21.91 21.81
	EB Poisson	0.11	0.09	0.21	20.20
	EB MaSEPTiDE	1.78	1.59 38.72	3.37 40.50	10.59 10.56

Long cascades		Time (seconds)			APE (%)
		Parameter estimation	Prediction	Total	
Sample 1	MaSEPTiDE	4.42	0.54 16.84	4.96 21.26	16.56 15.55
	EB Poisson	0.14	0.11	0.25	4.48
	EB MaSEPTiDE	4.89	1.86 37.63	6.75 42.52	2.75 2.20
Sample 2	MaSEPTiDE	2.94	0.52 17.03	3.45 19.96	20.68 19.91
	EB Poisson	0.12	0.09	0.21	6.71
	EB MaSEPTiDE	2.42	1.71 45.70	4.12 48.12	1.06 0.23
Sample 3	MaSEPTiDE	7.86	0.50 8.08	8.36 15.94	19.93 15.61
	EB Poisson	0.14	0.09	0.23	4.67
	EB MaSEPTiDE	4.70	1.46 42.01	6.16 46.71	4.94 3.81

Very long cascades		Time (seconds)			APE (%)
		Parameter estimation	Prediction	Total	
Sample 1	MaSEPTiDE	63.14	0.66 8.10	63.80 71.24	32.99 26.35
	EB Poisson	0.28	0.09	0.37	11.96
	EB MaSEPTiDE	44.87	1.53 19.49	46.40 64.36	4.69 5.02
Sample 2	MaSEPTiDE	13.55	0.58 19.54	14.13 33.09	6.98 7.12
	EB Poisson	0.19	0.08	0.27	6.15
	EB MaSEPTiDE	8.93	1.47 45.28	10.39 54.21	2.60 2.75
Sample 3	MaSEPTiDE	65.23	0.53 8.02	65.76 73.25	38.03 31.59
	EB Poisson	0.30	0.09	0.40	26.11
	EB MaSEPTiDE	14.75	1.50 41.57	16.25 56.31	17.99 18.96

While obtaining the parameter estimates and prediction values for efficient methods like the EB Poisson model using a local computer is convenient, models like the MaSEPTiDE model and the EB MaSEPTiDE model require the use of high performance computational clusters[2] to obtain the results swiftly. In particular, cascades of similar sizes can be grouped together to optimize the use of computational resources demanded. Network file transfer applications like `PuTTY` with various network protocol support, and clients like `FileZilla` are useful in running the codes and transferring files to and from the server with ease. For the whole test data set we have considered, assuming that a job consists of around 100 retweet cascades, running the jobs at six hours should suffice to yield the full results for the majority of the cascades, but a more conservative run time, say 12 hours, warrants a better completion rate.

By the APE values in each subtable of Table C.1.1, the EB Poisson model seems superior compared to the MaSEPTiDE model, and the EB MaSEPTiDE model is considerably more accurate in tweet popularity prediction than the original MaSEPTiDE model, a conclusion similar to that drawn based on the complete data set. The APEs based on the conditional expectations from the solve-the-equation and simulation-based approaches for the MaSEPTiDE model and the EB MaSEPTiDE model appear to be consistent with each other, as the predicted final popularity values using both approaches should be roughly equivalent.

C.2 Simulation Replications

We have mentioned that for the MaSEPTiDE model, predictions based on the simulation-based approach with sufficiently many replications should be consistent with those based on the solve-the-equation approach with adequate number of knots. To demonstrate this, we shall use the same samples of retweet cascades in Table B.2.1 consisting of varying final popularity values, and include the relevant results in Table C.2.1.

For the prediction tasks demonstrated in Table C.2.1 we note that the objective is to obtain the predicted popularity value from T to $\tilde{T}$, or $(N(\tilde{T}) - N(T))_{\text{pred}}$. This said, the columns in the table from left to right correspond respectively to the predicted value based on the solve-the-equation approach, the acceleration factor S used to inflate the simulated event numbers, the predicted values based on the simulation-based approach with and without the acceleration factor at both 50 and 100 replications, and finally the variance of simulated event numbers. The corresponding APEs have also been shown in Table B.2.1, to facilitate the evaluation of

[2] we have run our jobs using the Katana computational cluster under the settings of one node and one core per node, with 4GB of memory requested

Table C.2.1: Prediction results based on sample cascades of varying lengths at $T = 2$ hours. The column from left to right shows the prediction based on the solve-the-equation (STE) approach, the acceleration factor S, the mean number of events based on the simulation-based approach at 50 and 100 replications with and without the factor S, and finally the variance of the simulated event numbers. The simulation-based approach at 50 or 100 replications seems sufficient to yield a prediction consistent with that obtained using the solve-the-equation approach.

		STE	S	Mean × S		Mean		Variance	
				50	100	50	100	50	100
Very short cascades	Sample 1	7.22	1	7.44	7.42	7.44	7.42	7.19	6.99
	Sample 2	1.62	1	1.76	1.67	1.76	1.67	1.90	1.66
	Sample 3	22.93	6	24.36	22.50	4.06	3.75	4.06	3.62
Short cascades	Sample 1	0.11	1	0.16	0.12	0.16	0.12	0.22	0.15
	Sample 2	1.96	1	2.28	2.18	2.28	2.18	2.12	1.87
	Sample 3	32.63	8	34.72	32.08	4.34	4.01	5.45	4.49
Middle-length cascades	Sample 1	0.63	1	0.50	0.52	0.50	0.52	0.50	0.51
	Sample 2	39.54	10	43.00	40.80	4.30	4.08	4.01	3.08
	Sample 3	3.42	1	3.54	3.55	3.54	3.55	3.72	3.52
Long cascades	Sample 1	87.07	22	91.96	84.7	4.18	3.85	2.56	3.54
	Sample 2	99.78	25	103.00	98.0	4.12	3.92	3.82	3.51
	Sample 3	140.43	70	137.20	129.5	1.96	1.85	1.59	1.40
Very long cascades	Sample 1	645.55	323	613.70	584.63	1.90	1.81	1.52	1.37
	Sample 2	44.59	11	46.86	44.99	4.26	4.09	3.26	3.42
	Sample 3	458.29	229	435.10	416.78	1.90	1.82	1.52	1.36

the prediction performances for the MaSEPTiDE model under such scenario.

It can be seen from Table C.2.1 that the difference between the predicted values based on the solve-the-equation and simulation-based approaches becomes smaller in general as the number of simulation replications increases. A closer scrutiny further reveals that the variance and mean are very close to each other, and that 50 or 100 replications of the simulation are enough to guarantee a small relative error for cascades of varying intensity levels.

C.3 Ablation Studies

The models we have proposed, namely the MaSEPTiDE model, the EB MaSEPTiDE model, and the EB Poisson model, consist of component functions contributing to the predicted future popularity values, which directly affect the APEs. This said, it is beneficial to investigate the relative effect exerted by the individual component function from each of the model, a practice widely known as the ablation studies. If we take the MaSEPTiDE model for example, the empirical contribution of each component function in (4.1.3) can be inspected by letting $p(\cdot) = 1$ or $r(\cdot) = 1$.

We have mentioned in Section 4.4.2 that the solution of the functional equation used in the solve-the-equation approach to obtain the conditional expectation

is not always easy to obtain, and that the numerical integration may occasionally fail. By the excitation function with components in (4.1.3) and at $T = 2$ hours, the percentage of retweet cascades failing to obtain a legitimate solution from the solve-the-equation approach is 0.37%. The percentage rises to 0.50% when $p(\cdot)$ is dropped from the excitation function, and a staggering 2.10% when $r(\cdot)$ is dropped. Overall, dropping the infectivity function $p(\cdot)$ tends to worsen the model prediction performance, but dropping the impact function $r(\cdot)$ may have different effects. This is demonstrated in Table C.3.1, where the classes of cascade lengths are identical to those shown in Table B.2.1. The percentage of retweet cascades failing to obtain a solution based on the solve-the-equation approach for each combination of component functions has also been included in each subtable.

Table C.3.1: The changes in prediction performances based on dropping the individual component function, in accordance to the classes of retweet cascade lengths, based on the complete test data. The percentages of retweet cascades failing to return a solution based on the solve-the-equation approach have also been exhibited in each subtable. The infectivity function $p(\cdot)$ contributes more to accurate tweet popularity predictions.

Very short cascades	APE (%)			NA (%)
	Q_1	Q_2	Q_3	
$p(\tau)r(n)\phi(t-\tau)$	5.54	15.41	35.96	0.10
$r(n)\phi(t-\tau)$	5.77	16.65	44.69	0.15
$p(\tau)\phi(t-\tau)$	4.89	13.91	31.67	0.14

Short cascades	APE (%)			NA (%)
	Q_1	Q_2	Q_3	
$p(\tau)r(n)\phi(t-\tau)$	6.98	18.18	38.27	0.06
$r(n)\phi(t-\tau)$	7.66	20.50	46.32	0.09
$p(\tau)\phi(t-\tau)$	6.49	16.88	36.24	0.13

Middle-length cascades	APE (%)			NA (%)
	Q_1	Q_2	Q_3	
$p(\tau)r(n)\phi(t-\tau)$	8.20	20.69	42.01	0.08
$r(n)\phi(t-\tau)$	8.71	22.52	49.36	0.09
$p(\tau)\phi(t-\tau)$	7.50	19.16	40.84	0.23

Long cascades	APE (%)			NA (%)
	Q_1	Q_2	Q_3	
$p(\tau)r(n)\phi(t-\tau)$	8.69	22.36	45.72	0.07
$r(n)\phi(t-\tau)$	10.23	25.22	53.41	0.12
$p(\tau)\phi(t-\tau)$	8.62	21.56	47.42	0.38

Very long cascades	APE (%)			NA (%)
	Q_1	Q_2	Q_3	
$p(\tau)r(n)\phi(t-\tau)$	10.64	26.18	49.86	0.06
$r(n)\phi(t-\tau)$	13.80	32.15	59.29	0.05
$p(\tau)\phi(t-\tau)$	12.34	30.20	58.36	1.22

While dropping $p(\cdot)$ appears to always worsen the prediction performances, dropping $r(\cdot)$ seems to have variable effects depending on the lengths of the retweet cascades in question. This is manifested in Table C.3.1, where dropping $r(\cdot)$ seems to positively affect the prediction performances for all but very long cascades. However, the number of unsolvable functional equations will markedly increase when this component is absent from the excitation function, and so its inclusion is recommended. As a remark, the component functions can be changed to different forms accordingly for future models to suit to various needs. The effects are similar when changes are applied on the EB MaSEPTiDE model, but each change can be arduous since it involves the reconstruction of the prior distribution for the model parameters.

Another model that we have proposed, namely the EB Poisson model in (5.1.1), consists of two main components where $d(\cdot)$ can be dropped to see how it affects the overall prediction performance. As the circadian rhythms are more noticeable when the tweets considered are first published, it is relatively more sensible to demonstrate the effects at, say, time zero, as in Table C.3.2. Note that unlike the MaSEPTiDE

Table C.3.2: The changes in prediction performances based on dropping the component function $d(\cdot)$, in accordance to the classes of retweet cascade lengths, based on the complete test data. The function $d(\cdot)$ seems essential to make early popularity predictions more reliable.

Very short cascades	APE (%)		
	Q_1	Q_2	Q_3
$p(t)d(t)$	6.84	15.88	50.43
$p(t)$	6.98	16.16	51.90
Short cascades	**APE (%)**		
	Q_1	Q_2	Q_3
$p(t)d(t)$	15.32	25.67	36.61
$p(t)$	15.38	25.72	36.71
Middle-length cascades	**APE (%)**		
	Q_1	Q_2	Q_3
$p(t)d(t)$	24.15	40.95	50.74
$p(t)$	24.10	40.70	50.65
Long cascades	**APE (%)**		
	Q_1	Q_2	Q_3
$p(t)d(t)$	37.30	56.60	67.21
$p(t)$	37.53	56.69	67.12
Very long cascades	**APE (%)**		
	Q_1	Q_2	Q_3
$p(t)d(t)$	64.75	76.75	84.44
$p(t)$	64.81	76.78	84.36

model, the EB Poisson model is always able to produce a prediction. The APEs in Table C.3.2 reveal that dropping the function $d(\cdot)$ seems to only make popularity predictions slightly less accurate. Therefore, alternative forms of $d(\cdot)$ which incorporate more informative calendar effects, for instance the trends observable during the weekends and weekdays, might make popularity predictions even more accurate.

References

Agarwal, D., Chen, B.-C., and Elango, P. (2009). Spatio-temporal models for estimating click-through rate. In *Proceedings of the 18th international conference on World wide web*, pages 21–30. ACM.

Ahmed, M., Spagna, S., Huici, F., and Niccolini, S. (2013). A peek into the future: Predicting the evolution of popularity in user generated content. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 607–616.

Alves, R. A., Assunção, R., and de Melo, P. O. (2016). Burstiness scale: A highly parsimonious model for characterizing random series of events. *arXiv preprint arXiv:1602.06431*.

Aragón, P., Gómez, V., García, D., and Kaltenbrunner, A. (2017). Generative models of online discussion threads: State of the art and research challenges. *Journal of Internet Services and Applications*, 8(1):15.

Asur, S. and Huberman, B. A. (2010). Predicting the future with social media. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, volume 1, pages 492–499. IEEE.

Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. (2011). Everyone's an influencer: Quantifying influence on Twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 65–74. ACM.

Bandari, R., Asur, S., and Huberman, B. A. (2012). The pulse of news in social media: Forecasting popularity. *ICWSM*, 12:26–33.

Barabasi, A.-L. (2005). The origin of bursts and heavy tails in human dynamics. *Nature*, 435(7039):207–211.

Bollen, J., Mao, H., and Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, 2(1):1–8.

Bray, P. (2012). When is my tweet's prime of life?

Brooks, S. P. and Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of computational and graphical statistics*, 7(4):434–455.

Casella, G. (1985). An introduction to empirical Bayes data analysis. *The American Statistician*, 39(2):83–87.

Cha, M., Haddadi, H., Benevenuto, F., and Gummadi, P. K. (2010). Measuring user influence in Twitter: The million follower fallacy. *Icwsm*, 10(10-17):30.

Chen, F. and Hall, P. (2013). Inference for a nonstationary self-exciting point process with an application in ultra-high frequency financial data modeling. *Journal of Applied Probability*, 50(04):1006–1024.

Chen, F. and Hall, P. (2016). Nonparametric estimation for self-exciting point processes - a parsimonious approach. *Journal of Computational and Graphical Statistics*, 25(1):209–224.

Chen, F. and Stindl, T. (2018). Direct likelihood evaluation for the renewal Hawkes process. *Journal of Computational and Graphical Statistics*, 27(1):119–131.

Chen, F. and Tan, W. H. (2018). Marked self-exciting point process modelling of information diffusion on Twitter. *Ann. Appl. Statist.*, 12(4):2175–2196.

Cleveland, W. S. and Devlin, S. J. (1988). Locally weighted regression: An approach to regression analysis by local fitting. *Journal of the American Statistical Association*, 83(403):596–610.

Cowling, A. and Hall, P. (1996). On pseudodata methods for removing boundary effects in kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(3):551–563.

Crane, R. and Sornette, D. (2008). Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of Sciences*, 105(41):15649–15653.

Daley, D. J. and Vere-Jones, D. (2003). *An Introduction to the Theory of Point Processes Volume I: Elementary Theory and Methods*. Springer-Verlag, New York, 2nd edition.

Dassios, A., Zhao, H., et al. (2013). Exact simulation of Hawkes process with exponentially decaying intensity. *Electronic Communications in Probability*, 18(62):1–13.

DiNucci, D. (1999). Fragmented future. *Print*, 53(4):32–33.

Efron, B. (2010). *Large-Scale Inference: Empirical Bayes Methods for Estimation, Testing, and Prediction*. Institute of Mathematical Statistics monographs. Cambridge University Press, New York.

Engle, R. F. and Lunde, A. (2003). Trades and quotes: A bivariate point process. *Journal of Financial Econometrics*, 1(2):159–188.

Eysenbach, G. (2011). Can tweets predict citations? metrics of social impact based on Twitter and correlation with traditional metrics of scientific impact. *Journal of medical Internet research*, 13(4).

Famaey, J., Wauters, T., and De Turck, F. (2011). On the merits of popularity prediction in multimedia content caching. In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pages 17–24. IEEE.

Fan, J. (2018). *Local Polynomial Modelling and Its Applications: Monographs on Statistics and Applied Probability*. Routledge.

Feuerriegel, S. and Proellochs, N. (2018). *SentimentAnalysis: Dictionary-Based Sentiment Analysis*. R package version 1.3-2.

Figueiredo, F. (2013). On the prediction of popularity of trends and hits for user generated videos. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 741–746. ACM.

Filimonov, V. and Sornette, D. (2015). Apparent criticality and calibration issues in the Hawkes self-excited point process model: Application to high-frequency financial data. *Quantitative Finance*, 15(8):1293–1314.

Fletcher, R. (2013). *Practical Methods of Optimization*. John Wiley & Sons.

Fox, E. W., Short, M. B., Schoenberg, F. P., Coronges, K. D., and Bertozzi, A. L. (2016). Modeling e-mail networks and inferring leadership using self-exciting point processes. *Journal of the American Statistical Association*, 111(514):564–584.

Galuba, W., Aberer, K., Chakraborty, D., Despotovic, Z., and Kellerer, W. (2010). Outtweeting the Twitterers - predicting information cascades in microblogs. *WOSN*, 10:3–11.

Gao, S., Ma, J., and Chen, Z. (2015). Modeling and predicting retweeting dynamics on microblogging platforms. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 107–116. ACM.

Gneiting, T. (2011). Making and evaluating point forecasts. *Journal of the American Statistical Association*, 106(494):746–762.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(2009):12.

Golub, G. H., Heath, M., and Wahba, G. (1979). Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223.

Gupta, M., Gao, J., Zhai, C., and Han, J. (2012). Predicting future popularity trend of events in microblogging platforms. *Proceedings of the Association for Information Science and Technology*, 49(1):1–10.

Gürsun, G., Crovella, M., and Matta, I. (2011). Describing and forecasting video access patterns. In *INFOCOM, 2011 Proceedings IEEE*, pages 16–20. IEEE.

Han, B., Hui, P., Kumar, V. A., Marathe, M. V., Shao, J., and Srinivasan, A. (2012). Mobile data offloading through opportunistic communications and social participation. *IEEE Transactions on Mobile Computing*, 11(5):821–834.

Hasan, F. M., UzZaman, N., and Khan, M. (2007). Comparison of different POS tagging techniques (n-gram, HMM and Brills tagger) for Bangla. In *Advances and innovations in systems, computing sciences and software engineering*, pages 121–126. Springer.

Hawelka, B., Sitko, I., Beinat, E., Sobolevsky, S., Kazakopoulos, P., and Ratti, C. (2014). Geo-located Twitter as proxy for global mobility patterns. *Cartography and Geographic Information Science*, 41(3):260–271.

Hawkes, A. G. (1971). Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, pages 83–90.

Hawkes, A. G. and Oakes, D. (1974). A cluster process representation of a self-exciting process. *Journal of Applied Probability*, 11(3):493–503.

Hermida, A. (2010). Twittering the news: The emergence of ambient journalism. *Journalism practice*, 4(3):297–308.

Hong, L., Dan, O., and Davison, B. D. (2011). Predicting popular messages in Twitter. In *Proceedings of the 20th international conference companion on World wide web*, pages 57–58. ACM.

Hughes, A. L. and Palen, L. (2009). Twitter adoption and use in mass convergence and emergency events. *International Journal of Emergency Management*, 6(3-4):248–260.

Isaac, M. and Ember, S. (2016). For election day influence, Twitter ruled social media. *The New York Times*.

Java, A., Song, X., Finin, T., and Tseng, B. (2007). Why we Twitter: Understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM.

Kaplan, A. M. and Haenlein, M. (2010). Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53(1):59–68.

Kiwiel, K. C. (2001). Convergence and efficiency of subgradient methods for quasiconvex minimization. *Mathematical programming*, 90(1):1–25.

Kobayashi, R. and Lambiotte, R. (2016). TiDeH: Time-dependent Hawkes process for predicting retweet dynamics. In *Proceedings of the Tenth International AAAI Conference on Web and Social Media (ICWSM 2016)*, pages 191–200. Association for the Advancement of Artificial Intelligence.

Kong, S., Ye, F., and Feng, L. (2014). Predicting future retweet counts in a microblog. *Journal of Computational Information Systems*, 10(4):1393–1404.

Kumar, R., Mahdian, M., and McGlohon, M. (2010). Dynamics of conversations. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 553–562. ACM.

Kupavskii, A., Umnov, A., Gusev, G., and Serdyukov, P. (2013). Predicting the audience size of a tweet. In *ICWSM*.

Kwak, H., Lee, C., Park, H., and Moon, S. (2010). What is Twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, pages 591–600. ACM.

Lakkaraju, H. and Ajmera, J. (2011). Attention prediction on social media brand pages. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2157–2160. ACM.

Last, G. and Brandt, A. (1995). *Marked Point Processes on the Real Line: The Dynamic Approach*. Springer Science & Business Media.

Lee, C. and Wilkinson, D. J. (2018). A hierarchical model of non-homogeneous Poisson processes for Twitter retweets. *arXiv preprint arXiv:1802.01987*.

Lewis, P. A. and Shedler, G. S. (1979). Simulation of nonhomogeneous Poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413.

Li, C.-T., Shan, M.-K., Jheng, S.-H., and Chou, K.-C. (2016). Exploiting concept drift to predict popularity of social multimedia in microblogs. *Information Sciences*, 339:310–331.

Li, M., Wang, X., Gao, K., and Zhang, S. (2017). A survey on information diffusion in online social networks: Models and methods. *Information*, 8(4):118.

Liniger, T. J. (2009). *Multivariate Hawkes Processes*. PhD thesis, ETH Zurich.

Lymperopoulos, I. N. (2016). Predicting the popularity growth of online content: Model and algorithm. *Information Sciences*, 369:585–613.

Malandrino, F., Kurant, M., Markopoulou, A., Westphal, C., and Kozat, U. C. (2012). Proactive seeding for information cascades in cellular networks. In *INFOCOM, 2012 Proceedings IEEE*, pages 1719–1727. IEEE.

Malmgren, R. D., Stouffer, D. B., Motter, A. E., and Amaral, L. A. (2008). A Poissonian explanation for heavy tails in e-mail communication. *Proceedings of the National Academy of Sciences*, 105(47):18153–18158.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Martin, T., Hofman, J. M., Sharma, A., Anderson, A., and Watts, D. J. (2016). Exploring limits to prediction in complex social systems. In *Proceedings of the 25th International Conference on World Wide Web*, pages 683–694. International World Wide Web Conferences Steering Committee.

Matsubara, Y., Sakurai, Y., Prakash, B. A., Li, L., and Faloutsos, C. (2012). Rise and fall patterns of information diffusion: Model and implications. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 6–14. ACM.

Mehrdad, B. and Zhu, L. (2014). On the Hawkes process with different exciting functions. *arXiv preprint arXiv:1403.0994*.

Mishra, S., Rizoiu, M.-A., and Xie, L. (2016). Feature driven and point process approaches for popularity prediction. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1069–1078. ACM.

Moller, J. and Waagepetersen, R. P. (2003). *Statistical Inference and Simulation for Spatial Point Processes*. Chapman and Hall/CRC.

Morris, C. N. (1983). Parametric empirical bayes inference: Theory and applications. *Journal of the American Statistical Association*, 78(381):47–55.

Murthy, D. (2018). *Twitter*. Polity Press.

Naveed, N., Gottron, T., Kunegis, J., and Alhadi, A. C. (2011). Bad news travel fast: A content-based analysis of interestingness on Twitter. In *Proceedings of the 3rd International Web Science Conference*, page 8. ACM.

Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *The Computer Journal*, 7(4):308.

Newman, T. P. (2017). Tracking the release of IPCC AR5 on Twitter: Users, comments, and sources following the release of the working group i summary for policymakers. *Public Understanding of Science*, 26(7):815–825.

Nishi, R., Takaguchi, T., Oka, K., Maehara, T., Toyoda, M., Kawarabayashi, K.-i., and Masuda, N. (2016). Reply trees in Twitter: Data analysis and branching process models. *Social Network Analysis and Mining*, 6(1):26.

Ogata, Y. (1981). On Lewis' simulation method for point processes. *IEEE Transactions on Information Theory*, 27(1):23–31.

Ogata, Y. (1988). Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical association*, 83(401):9–27.

Oghina, A., Breuss, M., Tsagkias, M., and de Rijke, M. (2012). Predicting IMDb movie ratings using social media. In *European Conference on Information Retrieval*, pages 503–507. Springer.

O'Reilly, T. (2005). What is Web 2.0.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10, pages 1320–1326.

Petrovic, S., Osborne, M., and Lavrenko, V. (2011). RT to win! predicting message propagation in Twitter. *ICWSM*, 11:586–589.

Pinto, H., Almeida, J. M., and Gonçalves, M. A. (2013). Using early view patterns to predict the popularity of Youtube videos. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 365–374. ACM.

R Core Team (2016). R: A language and environment for statistical computing.

Rey, B. (2014). Your tweet half-life is 1 billion times shorter than carbon-14's.

Rizoiu, M.-A., Lee, Y., Mishra, S., and Xie, L. (2017). A tutorial on Hawkes processes for events in social media. *arXiv preprint arXiv:1708.06401*.

Roy, S. D., Mei, T., Zeng, W., and Li, S. (2013). Towards cross-domain learning for social video popularity prediction. *IEEE Transactions on multimedia*, 15(6):1255–1267.

Saad, D. (1998). Online algorithms and stochastic approximations. *Online Learning*, 5.

Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM.

Sheather, S. J. and Jones, M. C. (1991). A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 683–690.

Silverman, B. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis.

Simma, A. and Jordan, M. I. (2012). Modeling events with cascades of Poisson processes. *arXiv preprint arXiv:1203.3516*.

Szabo, G. and Huberman, B. A. (2010). Predicting the popularity of online content. *Communications of the ACM*, 53(8):80–88.

Tatar, A., de Amorim, M. D., Fdida, S., and Antoniadis, P. (2014). A survey on predicting the popularity of web content. *Journal of Internet Services and Applications*, 5(1):8.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., and Welpe, I. M. (2010). Predicting elections with Twitter: What 140 characters reveal about political sentiment. *ICWSM*, 10(1):178–185.

Utsu, T. (1961). A statistical study on the occurrence of aftershocks. *Geophys. Mag.*, 30:521–605.

Van Aelst, P., van Erkel, P., Dheer, E., and Harder, R. A. (2017). Who is leading the campaign charts? comparing individual popularity on old and new media. *Information, Communication & Society*, 20(5):715–732.

Vosoughi, S., Zhou, H., and Roy, D. (2016). Enhanced Twitter sentiment classification using contextual information. *arXiv preprint arXiv:1605.05195*.

Wu, B., Cheng, W.-H., Zhang, Y., and Mei, T. (2016). Time matters: Multi-scale temporalization of social media popularity. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 1336–1344. ACM.

Wu, B. and Shen, H. (2015). Analyzing and predicting news popularity on Twitter. *International Journal of Information Management*, 35(6):702–711.

Wu, S., Hofman, J. M., Mason, W. A., and Watts, D. J. (2011). Who says what to whom on Twitter. In *Proceedings of the 20th international conference on World wide web*, pages 705–714. ACM.

Xie, M. and Singh, K. (2013). Confidence distribution, the frequentist distribution estimator of a parameter: A review. *International Statistical Review*, 81(1):3 – 39.

Yan, Y., Tan, Z., Gao, X., Tang, S., and Chen, G. (2016). STH-Bass: A spatial-temporal heterogeneous bass model to predict single-tweet popularity. In *International Conference on Database Systems for Advanced Applications*, pages 18–32. Springer.

Yang, J. and Leskovec, J. (2011). Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 177–186. ACM.

Yu, S. and Kak, S. (2012). A survey of prediction using social media. *arXiv preprint arXiv:1203.1647*.

Zaman, T., Fox, E. B., and Bradlow, E. T. (2014). A Bayesian approach for predicting the popularity of tweets. *The Annals of Applied Statistics*, 8(3):1583–1611.

Zhang, P., Wang, X., and Li, B. (2013). On predicting Twitter trend: Factors and models. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 1427–1429. ACM.

Zhang, X., Fuehres, H., and Gloor, P. A. (2011). Predicting stock market indicators through Twitter. *Procedia-Social and Behavioral Sciences*, 26:55–62.

Zhao, Q., Erdogdu, M. A., He, H. Y., Rajaraman, A., and Leskovec, J. (2015). SEISMIC: A self-exciting point process model for predicting tweet popularity. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1513–1522. ACM.

Zhu, L. (2013). *Nonlinear Hawkes Processes*. PhD thesis, Citeseer.

Index