Robert Gordon University

Doctoral Thesis

An Association Rule Dynamics and Classification Approach to Event Detection and Tracking in Twitter

Supervisors:

Dr. Mohamed Gaber

Dr. Frederic Stahl

Author:

Mariam Adedoyin-Olowe

A thesis submitted in partial fulfilment of the requirements

of the Robert Gordon University for the degree of

Doctor of Philosophy

in the

Institute for Innovation, Design and Sustainability

Computing Science

May 2015

“. . . “Most breaking news stories are often posted on Twitter network before they are reported on traditional news media. This can be attributed to the presence of Twitter subscribers at locations of events who are always eager to post on the network in (near) real-time before the arrival of renowned newsagents.”.

Mariam Adedoyin-Olowe

ROBERT GORDON UNIVERSITY (IN BLOCK CAPITALS)

Abstract

School of Computing Science and Digital Media

Computing Science

Doctor of Philosophy

An Association Rule Dynamics and Classification Approach to Event Detection and Tracking in Twitter

by Mariam Adedoyin-Olowe

Twitter is a microblogging application used for sending and retrieving instant on-line messages of not more than 140 characters. There has been a surge in Twitter activities since its launch in 2006 as well as steady increase in event detection research on Twitter data (tweets) in recent years. With 284 million monthly active users [1] Twitter has continued to grow both in size and activity. The network is rapidly changing the way global audience source for information and influence the process of journalism [Newman, 2009]. Twitter is now perceived as an information network in addition to being a social network. This explains why traditional news media follow activities on Twitter to enhance their news reports and news updates. Knowing the significance of the network as an information dissemination platform, news media subscribe to Twitter accounts where they post their news headlines and include the link

[1] https://about.twitter.com/company

to their on-line news where the full story may be found. Twitter users in some cases, post breaking news on the network before such news are published by traditional news media. This can be ascribed to Twitter subscribers' nearness to location of events.

The use of Twitter as a network for information dissemination as well as for opinion expression by different entities is now common. This has also brought with it the issue of computational challenges of extracting newsworthy contents from Twitter noisy data. Considering the enormous volume of data Twitter generates, users append the hashtag *(#)* symbol as prefix to keywords in tweets. Hashtag labels describe the content of tweets. The use of hashtags also makes it easy to search for and read tweets of interest. The volume of Twitter streaming data makes it imperative to derive Topic Detection and Tracking methods to extract newsworthy topics from tweets.

Since hashtags describe and enhance the readability of tweets, this research is developed to show how the appropriate use of hashtags keywords in tweets can demonstrate temporal evolvements of related topic in real-life and consequently enhance Topic Detection and Tracking on Twitter network. We chose to apply our method on Twitter network because of the restricted number of characters per message and for being a network that allows sharing data publicly. More importantly, our choice was based on the fact that hashtags are an inherent component of Twitter.

To this end, the aim of this research is to develop, implement and validate a new approach that extracts newsworthy topics from tweets' hashtags of real-life topics over a specified period using Association Rule Mining. We termed our novel methodology Transaction-based Rule Change Mining *(TRCM)*. *TRCM* is a system built on top of the Apriori method of Association Rule Mining to extract patterns of Association Rules changes in tweets hashtag keywords at different periods of time and to map the extracted keywords to related real-life topic or scenario. To the best of our knowledge, the adoption of dynamics of Association Rules of hashtag co-occurrences has not been explored as a Topic Detection and Tracking method on Twitter. The application of Apriori to hashtags present in tweets at two consecutive period t and $t + 1$ produces two association rulesets, which represents rules evolvement in the context of this research. A change in rules is discovered by matching every rule in ruleset at time t with those in ruleset at time $t + 1$. The changes are grouped under four identified rules namely 'New' rules, 'Unexpected Consequent' and 'Unexpected Conditional' rules, 'Emerging' rules and 'Dead' rules. The four rules represent different levels of topic real-life evolvements. For example, the emerging rule represents very important occurrence such as breaking news, while unexpected rules represents unexpected twist of event in an on-going topic. The new rule represents dissimilarity in rules in rulesets at time t and $t+1$. Finally, the dead rule represents topic that is no longer present on the Twitter network. TRCM revealed the dynamics of Association Rules present in tweets and demonstrates the linkage between the different types of rule dynamics to targeted real-life topics/events.

In this research, we conducted experimental studies on tweets from different domains such as sports and politics to test the performance effectiveness of our method. We validated our method, TRCM with carefully chosen ground truth. The outcome of our research experiments include:

- Identification of 4 rule dynamics in tweets' hashtags namely: New rules, Emerging rules, Unexpected rules and 'Dead' rules using Association Rule

Mining. These rules signify how news and events evolved in real-life scenario.

- Identification of rule evolvements on Twitter network using *Rule Trend Analysis* and *Rule Trace.*

- Detection and tracking of topic evolvements on Twitter using Transaction-based Rule Change Mining *TRCM.*

- Identification of how the peculiar features of each *TRCM* rules affect their performance effectiveness on real datasets.

Acknowledgements

This research project could not have seen the light of the day without the help and support of many people. In particular, I would like to acknowledge the contributions and support of a good number of outstanding figures.

Foremost, I would like to offer sincere appreciation to my first supervisor, Dr. Mohamed Medhat Gaber. His time, knowledge, help, responsiveness, encouragement and mentorship throughout my PhD programme are sincerely appreciated.

I would also like to thank my second supervisor, Dr Frederic Stahl, for his time, knowledge and expert critique. Special thanks goes to Dr. Carlos Martins Dancausa for his selfless help and keen interest in my work since the first time he listened to my presentation couple of years ago at a research seminar in RGU. Special thanks also goes to Dr. Philip De Groot and Dr. João Bártolo Gomes for their immeasurable expert assistance and advice on R programming throughout my PhD journey.

My heart-felt thanks go to my husband Adedoyin Olowe for his unending support and great encouragement. To my two loving children, Tade and Oreoluwa, thanks for being well-behaved and supportive throughout my PhD years. I would also like to thank my parents Mr & Mrs Balogun for their prayers and frequent phone calls. Many thanks to all my fellow PhD colleagues, whose friendship has made this process much easier.

Contents

List of Figures

List of Tables

This work is dedicated to the Glory of God and to my loving trio Oyin, Tade and Ore.

Chapter 1

Introduction

1.1 Introduction

Twitter has been reported to have the highest number of users in its rank of microblogging applications [Bizer et al., 2012, Chakrabarti and Punera, 2011]. Traditional newsagents closely monitor the activities on the network to enhance the contents of their news broadcast [Evans, 2010]. Other entities such as individuals, business organisations and government bodies are relying on tweets (Twitter data) posted on the Internet for decision-making. Twitter users generate high volume of streaming data on the web. This leads to difficulty in computational data retrieval from Twitter. However, to enhance the readability of tweets and to describe their contents, users label prominent keywords in tweets with the hashtag *(#)* symbol as prefix. Twitter is known to be the first social network to introduce hashtag-click that enables a user to navigate to other tweets that include the same hashtag. Appropriate hashtag labelling of tweets does not only increase their readability, but also enhance retrieval of such tweets from those that constitute noise on the network. Simply put, noisy tweets are the irrelevant tweets on Twitter network. According to Laniado & Mika [Laniado and Mika, 2010], hashtag specification is the most effective way of extracting tweets of a particular topic on Twitter.

1.2 Benefits of Hashtag on Twitter

News and event-centred data sharing and searching on Twitter has increased in recent times. Users can classify and search for tweets of interest by hashtags. Appropriate use of hashtag can result in discussion groups for specific topic [Feng and Wang, 2014]. Hashtags enable the detection of emerging topic on Twitter. For news media, it enables opportunity to report the news to the world. Whereas, for business, it allows opportunity to know the latest things people are saying about their company or product. Timely detection of crucial emerging tweets can assist business in making swift and important decisions. Businesses create official hashtag *(#)* for the prospect of connecting with their stakeholders as well as for implementation of branding strategies. Hashtag increases the social presence of businesses and allow users to join in specific conversation on the network. Individual users detect and track topics on the network by using related hashtag(s). Hashtag provides control for query extension during significance review [Efron, 2010]. It also makes topical tweets more visible, for example $\#USElection2012$ was widely used during US elections in 2012. Scientific communities create official hashtags for communicating with their community of users. They include hashtag to the tweets they post on Twitter in order to enhance its readability. Government organisations use hashtags to pass information to the populace. Educational institutions are also known to prefix their institutional name with the # symbol to make it easy for them to view tweets posted about them and to enhance the readability of their own tweets posted on-line. It has become common practice on social network to have a list of trending hashtags that presents latest news and widely discussed topics in real-time. Hashtag helps businesses to interact with their customers and attract new ones. Individual and organisational presence on Twitter has increased in the last decade and the same applies to users' reliance on the network for information and decision-making.

1.3 The Increase in Twitter Reliance

Twitter posts are used to publicise topics and events that may result in public discussion or awareness, or even both. Individuals and other entities are relying more on information posted on Twitter for decision-making [Verma et al., 2011]. Users visit the network to either post or read tweets posted by other users. Tweets posted on-line include breaking news, local and global events, entertainments news, sports news, business news and celebrity gossips. The network offers its users the opportunity of sharing and receiving news/information in real-time without location restriction. The advent of smart devices has greatly increased the volume of tweets posted per second. Traditional newsagents follow the activities on Twitter to assist them in updating related news [Newman, 2009]. The Increase in user activities on Twitter network has necessitated the application of computational methods of crawling, retrieving, analysing and extracting newsworthy topics/events from the network.
Many Topic Detection and Tracking **(TDT)** experiments have been conducted on Twitter data [Adedoyin-Olowe et al., 2013,Agarwal et al., 2012,Becker et al., 2011,Becker et al., 2012,Mathioudakis and Koudas, 2010,Naaman et al., 2011, Osborne et al., 2012,Phuvipadawat and Murata, 2010,Sakaki et al., 2010,Tumasjan et al., 2010,Watanabe et al., 2011]. The results of these experiments show the dynamism of Twitter network as information dissemination tool and the efficiency of data mining methods in extracting newsworthy contents from

tweets posted on-line. Twitter is known to have played a major role in publicising and sustaining the trend of major news of global relevance on the network. Events like the 2011 Egyptian political uprisings [Starbird and Palen, 2012] and the US Elections 2012 [Wang et al., 2012] were widely tweeted given that Twitter users did not only follow the trend of these events, but also contributed to the topic by expressing their opinion.

1.4 TDT Challenges in Twitter

Twitter network is known to be the most popular microbloging application on social network sites since its launch in 2006 [Grosseck and Holotescu, 2008, Pak and Paroubek, 2010]. It offers its users the opportunity of posting and receiving instantaneous information from the network. It also allows users the prospect of following other users on the network and gaining access to their on-line tweets from their own update feed. Followers sometimes re-tweet (re-post) contents posted by their following users as well as contribute to the topic. The network record about 500 million tweets per day, with 80% of them sent via mobile devices [1]. Traditional news media follow the activities on Twitter network in order to retrieve interesting tweets that can be used to enhance their news reports and news updates.

Most breaking news stories are often posted on the network before they are reported on traditional news media. This can be attributed to the presence of Twitter subscribers at locations of events who are always eager to post on the network in (near) real-time before the arrival of renowned newsagents. However, retrieving information manually from the network can be likened to "looking for a needle in a haystack". While some on-line tweets relay credible information,

[1] https://about.twitter.com/company

others are implausible and contribute only noise when retrieving information from the network.

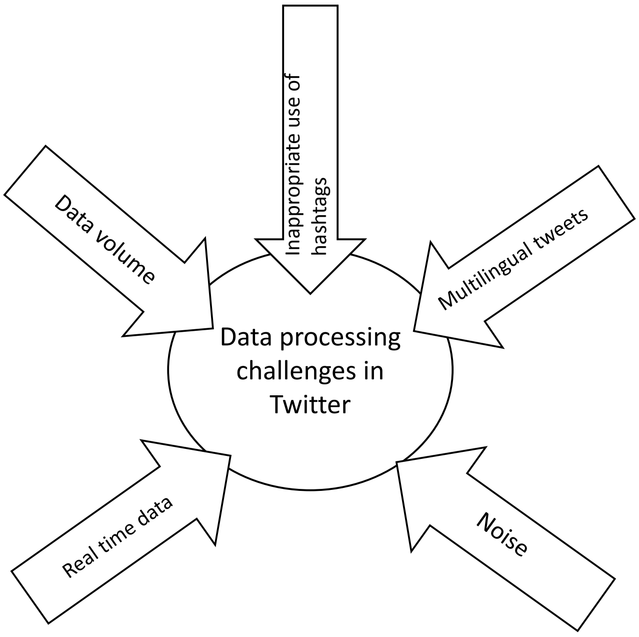

Inappropriate use of hashtags
Data volume
Multilingual tweets
Data processing challenges in Twitter
Real time data
Noise

FIGURE 1.1: Data processing challenges in Twitter

Fig. 1.1 depicts challenges on Twitter. **Noise** on Twitter is mostly caused by irrelevant posts and **spam** messages. Spam are unsolicited contents (tweets, urls, images or follower fraud) on the network. Spam continue to increase as Twitter continue to grow. Spammers derive different unlawful ways of posting contents on-line in the attempt to attract other users to read the posts [Yardi et al., 2009]. This includes the use of short urls linking unsuspecting users to malware or spurious site. Spammer sometimes exploit a trending hashtag by incorporating it into their own hashtag keywords. By so doing the chances of their spam tweets being visible on the network is increased. Some Twitter users seek followership on the network for commercial gain (follower fraud) by selling the list of their 'followers' details to businesses for money. On the other hand, genuine influencers are known for posting and spreading quality information on the network [Bakshy et al., 2011] and enhancing the opinion and decision-making of followers and other entities [Pang and Lee, 2008]. Like Twitter, other popular on-line networks are being negatively affected by spammers [Shirky, 2004]. **Real-time data** stream also pose a challenge on Twitter. Crawling relevant tweets and classifying them in real-time presents a more complete situation than in the case of non-real-time network [Dong et al., 2010]. Twitter is known as a network for *big data* generation [Bollier and Firestone, 2010]. The huge **volume of data** generated on Twitter on a daily basis makes it impossible to manually extract newsworthy tweets from the network. The problem of **multilingual tweets** relating to the same topic also poses a challenge on Twitter. Analysing global topic becomes complex when tweets are posted in different languages. This compels analysts to consider only the ones posted in language familiar to them [Krishnamurthy et al., 2008] and in the process discarding those that are likely to be newsworthy. The experiments conducted in the research reported in this thesis also face this challenge as only English tweets were considered for the experiments. The dynamics of Twitter compounds the computation of its data and requires efficient data mining techniques like Topic Detection and Tracking *(TDT)* to extract useful contents from the network. Twitter streams **real-time data** which are generated constantly and resulting in classification problems [Bifet and Frank, 2010].

This research focuses on the challenge of **inappropriate use of hashtag** in tweets. Annotating tweets with an unambiguous hashtag leads to better user experience. Retrieval of topical tweets from the high volume of Twitter data will be easier. We are motivated to consider the challenge of inappropriate use of hashtag in on-line tweets because of the importance and benefits that can be derived from the right use of hashtag on Twitter as explained in Section 1.2.

TDT experiments on Twitter data is pertinent because of the relevant information embedded in the hundreds of millions tweets posted on the network globally every second of the day. Although *TDT* research are being conducted to solve computational challenges on Twitter in the last decade, very limited

experiments have been targeted toward the mapping of hashtag keywords in tweets to real-life topics and events. Experts in the field have conducted experiments on event detection [Weng and Lee, 2011, Sakaki et al., 2010, Cataldi et al., 2010, Vieweg et al., 2010] events classification [Becker et al., 2011], events clustering and events tracking on Twitter network.
Considering the importance of hashtag on Twitter, we are motivated to carry out comprehensive empirical investigations of extracting frequent co-occurring hashtag keywords in tweets over a specified consecutive period and mapping our results to related real-life topics/events.

1.5 Research Aims and Objectives

To this end, this research is developed to exploit the presence of hashtag in tweets for topic detection and classification applications. This is achieved by conducting experiments revealing how the appropriate hashtag labelling of tweets can demonstrate temporal evolvements of related real-life topic/event. The investigation is carried out using **Association Rule Mining** *(ARM)* and our novel methodology termed **Transaction-based Rule Change Mining** *(TRCM)*. *TRCM* framework was built using *Apriori* method of *ARM*. The system defines patterns of *ARs* changes in tweets at different periods in relation to similar real-life scenario. To build *TRCM* system, the left hand side *lhs/conditional* and the right hand side *rhs/consequent* parts of rules in *Apriori* are employed to analyse hashtags present in tweets. Evaluation of the *lhs* and the *rhs* is used to identify the *ARs* present in tweets at different time. The similarities and differences in the *AR* in the rulesets r_i^t and r_j^{t+1}, (where t is the time and i, j are rules present in tweets at t and $t+1$ respectively) are measured to determine *TRCM* rules namely; “Emerging”, “Unexpected”, “New” and “Dead” rules in tweets. As demonstrated in Fig. 1.2.

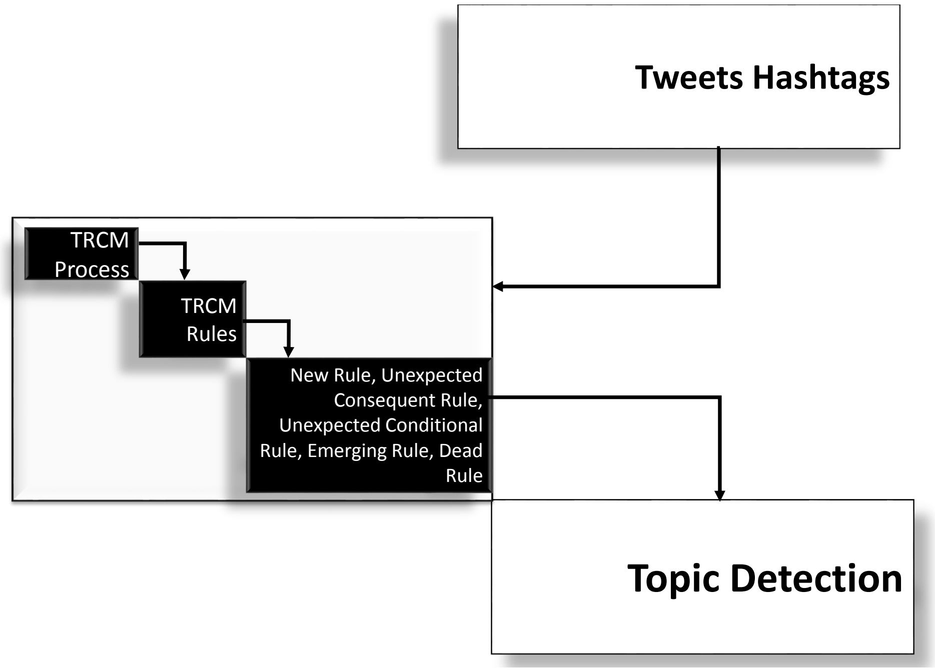

Tweets Hashtags
TRCM Process
TRCM Rules
New Rule, Unexpected Consequent Rule, Unexpected Conditional Rule, Emerging Rule, Dead Rule
Topic Detection

FIGURE 1.2: Simple TRCM Process

The aims of our research are:

- To detect newsworthy events/topics from Association Rules *(ARs)* present in tweets' hashtags at 2 consecutive period of time using a novel methodology termed **Transaction-based Rule Change Mining (TRCM)**;
- to map hashtag keywords present in detected *ARs* to related real-life topics/events;
- to track the evolvments of real-life topic/event in news broadcast of traditional newagents using *TRCM* and;
- to show the relationship between hashtag keywords and *ARs* identified by *TRCM* in their related real-life topic/event using visualisation.

To achieve our research aims, the following steps will be employed:

- The development of a set of techniques that are capable of extracting and detecting newsworthy events/topics from Association Rules present in tweets' hashtags at 2 consecutive period.
- The mapping of detected *ARs* in tweets' hashtags to related real-life events/topics.
- The application of *TRCM* to tweets from different domains to establish how the dynamism of event development can affect the performance of *TRCM*.
- The presentation of real-life case studies from diverse domains to validate the conducted experiments on topic/event tracking.
- The visualisation of *ARs* rules identified in research case studies.
- The presentation of the experimental results, recommendations as well as possible future work.

1.5.1 Research Contributions

The contributions of the research conducted in this thesis are highlighted as follows:

- Identification of 4 rule dynamics in tweets' hashtags that typifies news and events evolvements in real-life scenario. These identified rules dynamics are namely: "Emerging Rule", "Unexpected Consequent Rule & Unexpected Conditional Rule", "New Rule" and "Dead Rule".
- Detection and tracking of real-life topics/events from hashtag labels present in on-line tweets.

- Analysis of **Rule Trend** and **Rule Trace** in on-line tweets. This analysis reveals how the evolvements of real-life news are replicated in on-line tweets allowing users to trace back the origin of real-life news.

- Measurement of *TRCM* performance variation on tweets from different domains by carrying out analysis on tweets from diverse domains.

1.6 Organisation of the Thesis

This section gives a brief overview of the rest of the works completed in the research and their organisation in this thesis.

- Chapter 2 gives a comprehensive survey on Twitter Network Analytics. We explore some of the notable work already conducted on the network and its data, especially those conducted in the area of topic detection and tracking.

- Chapter 3 gives an overview of Association Rule Mining *(ARM)* and its measures such as the support, confidence and lift. It also justifies the adoption of the *Apriori* method of *ARM* for the research experiments.

- Chapter 4 proposes Transaction-based Rule Change Mining *(TRCM)*, the research methodology used in this thesis. The chapter explains the different *TRCM* tools such as *Rule Similarities and Difference*, *Rule Matching*, *TRCM* Rules definitions and how the rules evolve. It also demonstrates the *Rule Trend Analysis* as well as how rules evolve from one *Time Frame Window (TFW)* to the next during their life span on Twitter network.

- Chapter 5 demonstrates a quantitative experimental approach to event detection and tracking with its application to sports and politics.

- Chapter 6 demonstrates qualitative study of *TRCM* for topic/event tracking using **Rule Type Identification - Mapping** *(RTI-mapping)* on five real-life datasets from four different domains namely; politics, social, socioeconomic and business. The chapter explains the concept of the *"TwO" - "NwO" state.* This state demonstrates how topic/event tweets and related real-life news broadcast by traditional newsagents can be interrelated and subsequently compared to corroborate the authenticity of Twitter posts. We present the visualisation of one of the experimental datasets using our novel **TRCM-Viz** built with *NoSQL Neo4j* to demonstrate the relationship between the hashtags and the rules (nodes).
- Chapter 7 concludes the thesis with discussions and possible future work.

Chapter 2

A Survey of Twitter Network Analytics

2.1 Social Network Data

Twitter forms part of Internet-based applications known as the **Social Network Sites**. Social Network **(SN)** benefited from the concept and technology of Web 2.0 which enables the creation and interchange of user generated content [Kaplan and Haenlein, 2009]. *SN* can simply be referred to as the media used to be social [Safko, 2010]. *SN* sites are known for big data generation, which gives rise to computational challenges and complexities. Everyday Internet users visit different *SN* sites either to post or to retrieve information. *SN* users post contents pertaining to their personal lives (on Facebook and Twitter) some post pictures (on Instagram) while others post videos (on YouTube, Facebook and Twitter). Users also create personal blogs where they post real issues for discussion with other users or with the public. People follow and read personal blogs of experts in certain fields. Writing on specialised blogs often receive wide readability from community of other experts in that field.

SN sites generate data in real-time, which contributes immensely to the rapid growth of global databases. As of 2011 the world database is known to generate 2.5 quintillion bytes of data daily [Henno et al., 2013]. As of 2014 data generated every minute on some popular social networks (as presented in Fig 2.1) are known to contribute immensely to the global database. **Twitter** users post over 347, 000 tweets on-line [1]. *Google*$^+$ [2] users are reported to upload over 140, 000 every minutes. **YouTube** users upload 300 hours of videos [3], while **Flickr** users upload over 1, 270 [4]. **Instagram** users post over 48, 000 photos [5].

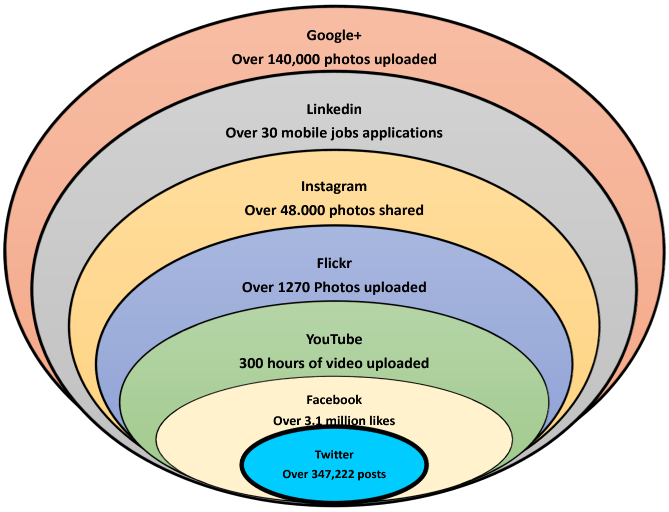

Google+
Over 140,000 photos uploaded
Linkedin
Over 30 mobile jobs applications
Instagram
Over 48.000 photos shared
Flickr
Over 1270 Photos uploaded
YouTube
300 hours of video uploaded
Facebook
Over 3.1 million likes
Twitter
Over 347,222 posts

Figure 2.1: Data Generated on Some SN Sites Every Minute

Facebook is said to record over 3.1 million likes, while Linkedin users use mobile

[1] (http://goo.gl/CQ7WST)
[2] (http://bootcampdigital.com/10-google-stats-that-will-blow-your-mind/)
[3] (http://www.adweek.com/socialtimes/files/2014/06/social-media-statistics-2014.jpg)
[4] (https://www.flickr.com/photos/franckmichel/6855169886/)
[5] (https://instagram.com/press/)

device to post over 30 job applications [6]. In addition to the statistics given, *SN* sites are known to generate data every second (in real-time), requiring up-to-date techniques to mine and store data in a contextualised format so that it can be easily analysed as required.

Figure 2.2: Common Social Media.

Like other *SN* sites shown in Fig. 2.2, it is evident that Twitter data is increasing at a rapid rate [Lasorsa et al., 2012] and as data increases, human knowledge about data decreases. To this end, there is an imperative need for a more precise and easy way to decipher techniques of retrieving constructive, valid and understandable results from big data. Many big organisations are now employing more sophisticated tools to handle their ever-growing database in order to extract useful data that can enhance their organisational decision and policy making. Individuals and organisations rely on *SN* as one of the means of communicating with their audience. The high rate of acquisition and the use of personal computers and other sophisticated smart mobile devices by people

[6] (http://expandedramblings.com/index.php/linkedin-job-statistics/)

around the world has aided *SN* sites to stream large-scale data. Telecommunications, electronic mail and electronic messengers like Skype, Yahoo messenger, Google Talk and MSN Messenger are also considered as *SN* [Aggarwal, 2011]. Local events on social media hardly remain local nowadays, as users are quick to post interesting events on the Internet (especially on Twitter), turning local events into issues for global discussions. The majority of mobile phone users in the world today use their phones to connect to the Internet more than using it for the primary purpose of making and receiving calls/text messages. 80% of tweets posted on-line are reported to be sent from smart mobile devices [7], as mentioned in Section 1.4. Many retail stores now include on-line stores to their chains and encourage buyers to leave reviews and/or 'Likes' on products/services they have experienced on popular social media sites, thereby adding to the size of data generated on-line. *SN* has contributed significantly to the success stories of many popular big businesses [Kaplan, 2012] in so many ways.

2.1.1 The Power of Social Networks

SN has endowed consumers and other stakeholders with unimaginable power of participation in businesses they have stake in [Evans, 2010, Parameswaran and Whinston, 2007]. More people are now relying on information given on *SN* when deciding on products/services to buy, film to watch at the cinema or school to enroll in [Pang and Lee, 2008]. To a large extent this information helps to eradicate the possibility of more people making the same mistake for lack of information [Qualman, 2012]. Reviewing products and services on-line is also a means of compelling businesses to improve on their products and services since high percentage of patronage is often derived based on reviews of other customers. Big businesses devote time to filter big data generated from *SN*

[7] https://about.twitter.com/company

sites to make valuable decisions. Research on extracting top quality information from social media [Agichtein et al., 2008,Liu et al., 2009] and on presenting Twitter events content is gaining attention lately [Long and Yu, 2009,Shamma et al., 2010]. Events features can be used to develop query-design approaches for retrieving content associated with an event on different social network sites. These approaches discover ways of using event content detected on one social media to retrieve more information on other social media sites [Becker et al., 2012]. Business organisations utilise mobile *SN* to conduct marketing research, sales promotions/discounts, communication and relationship development/loyalty programs. This is presented in four pieces of advice for mobile social networks usage termed the Four I's of mobile social media. The four I's are namely; *integration* of their mobile social media activities into the lives of users, *individualisation* of activities that takes into account the preferences of each user's *involvement* by way of conversation, and *initiation* of the creation of user-generated content and word-of-mouth [Kaplan, 2012].

2.2 Twitter Network Analytics

The surge in Twitter activities [Li et al., 2014] since its launch in 2006 as well as the steady rise in event detection awareness on the network [Lau et al., 2012] in recent times has continued to attract research on the network and its data. With 288 million monthly active users as of the last quarter of 2014 [8], Twitter continues to develop both in size and activity. The network is rapidly changing the way social networks audience around the world explore information and influence the process of journalism [Lasorsa et al., 2012,Newman, 2009]. Twitter is becoming more of an information network rather than just a social network when compared with other social networks like Facebook and Tumblr. This explains why traditional news media follow activities on Twitter to enhance their

[8] https://about.twitter.com/company

news reports and updates. News media cite URL that contain full story they broadcast on their Twitter page and by that means enhance the readability of their news broadcast. Breaking news is sometimes posted on Twitter before they are published by traditional news media due to users' geographical nearness to location of events [Castillo et al., 2011, Cataldi et al., 2010]. The dynamic and streaming nature of Twitter data (known as tweet) is characterised with noise resulting in the difficulty of manual extraction of meaningful contents from the network. Where some tweets are relevant to specific real-life event and are worthy of being extracted, others constitute noise to the network [Naaman et al., 2010]. This shows the need for filtering to extract relevant tweets from Twitter. According to Allan [Allan, 2002a], topic as defined in Topic Detection and Tracking (TDT) context can be "*a set of news stories that are strongly related by some similar real-world events*". Event often triggers topic; for instance, breaking news announcing the winner of the US presidential elections will trigger other related news such as the President's victory speech and reports on congratulatory message from opposition candidate. All these unfolding events will generate news updates resulting in the evolvement of related topics. TDT methods are currently used to detect and track trending events on Twitter over time [Aiello et al., 2013].

The importance of detecting the unfolding of significant patterns in tweets using both statistical methods and computational programming for performance measurement has continued to increase. *(TDT)* challenges on Twitter streaming data as discussed in Section 1.4 has made it pertinent to design computational ways of mining and analysing data generated on the network for meaningful use by its users. Research on Twitter analytics has gained popularity as authors from diverse fields are analysing the network from different perspectives ranging from opinion mining/sentiment analysis, influencer on the network to TDT (which is the area considered in this thesis). Data mining methods are being applied to Twitter data for the purpose of extracting contents that are related to specific real-life topics, events or issues. Survey of related work in Twitter analytics is discussed in subsequent sections of this chapter.

2.3 Opinion Mining and Sentiment Analysis on Twitter

Twitter network is a viable platform for unrestricted public opinion and sentiment expression. People visit the site to post their opinion of diverse topics ranging from local, national to global issues. Twitter (and other social network sites) have offered social power (and responsibilities) on users by way of saying what they want to say, how they want to say it and to whom they want to say it to, without being restricted by location. Most opinions expressed on Twitter are considered valuable by entities concerned. Such postings can be used for decision-making. Messages shared on-line by users can be used for opinion/sentiment analysis task [Pak and Paroubek, 2010].

Users device emotion icons to describe how they feel without having to spell it out in plain texts. Most of the emotion icons are common to different social networks. The relevance of opinion expressed on Twitter and the importance attached to them by businesses and other organisations including government bodies has increased the need for computational analysis of the opinion. The experiments in Kouloumpis et al [Kouloumpis et al., 2011] show that Part-Of-Speech (POS) structures may not be important for opinion analysis in Twitter, whereas, structures from obtainable sentiment lexicon (sample shown in Fig. 2.3) can be relevant when used jointly with emoticon of positive, negative and neutral. Bollen et al [Bollen et al., 2011b] analyse the text content of day-to-day Twitter Feeds using *OpinionFinder* and *Google-Profile of Mood States* that quantifies mood in terms of 6 scopes namely calm, alert, sure, vital, kind, and happy). Using time series of the moods, they compared their performance in detecting public's reaction to the presidential election and Thanksgiving Day in 2008. Finally, they employed Granger causality analysis and a Self-Organizing

Fuzzy Neural Network to inspect the theory that public mood states, as measured by the OpinionFinder and GPOMS mood time series, are predictive of changes in Dow Jones Industrial Average (DJIA) closing values. Their results showed the significance of DJIA could be improved by including only precise public mood scopes.

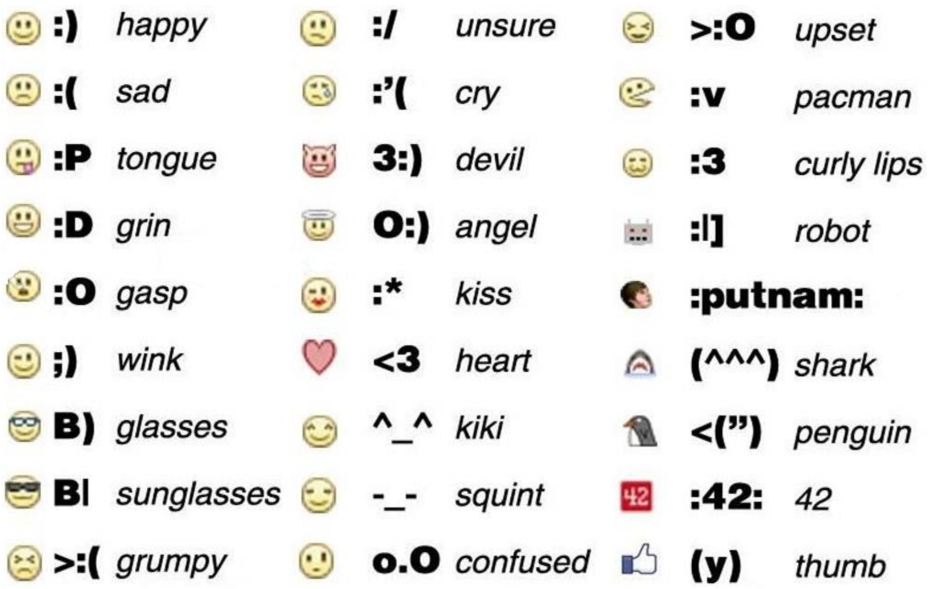
Furthermore, in Bollen et al [Bollen et al., 2011a] they quantify the sentiment of tweets using comprehensive version of the Profile of Mood States (POMS). They applied psychometric instrument to extract another set of six mood states (tension, depression, anger, vigour, fatigue, confusion) from the combined Twitter content and calculate a six-dimensional mood vector in the time-line on a daily basis. The results obtained were matched to the time-line of famous events that occurred during the same period. They argue that sentiment analysis of minute text corpora like tweets is best achieved via a syntactic, term-based approach without any machine learning training requirements. On the other hand, the work of Conover et al [Conover et al., 2011] applied support vector machine SVM to predict the political orientation of Twitter users based on the content and system of their political message in the US mid-term election in 2010. It was established that the application of latent semantic analysis to users' tweets contents was able to detect concealed structure in tweets that are connected with political association. However, it was resolved that topic detection did not enhance prediction performance.

Similarly, Anjaria et al [Anjaria and Guddeti, 2014] aimed to exploit the influence factor to predict the outcome of the US Presidential Elections 2012 and Karnataka Assembly Elections 2013. They proposed a hybrid approach of mining opinion using direct and indirect structures of tweets based on SVM, Naive Bayes, Maximum Entropy and Artificial Neural Networks based supervised classifiers. The work of Diakopoulos and Shamma [Diakopoulos and Shamma, 2010] revealed an analytical approach comprising visual illustrations and metrics used for understanding of the sentiment embedded in tweets related to televised political debate between two aspirants during the US Elections in 2008. They established that sentiments can be detected by observing the anomalies in the pulse of the sentiment indicator, while contentious topics can be detected by observing associated sentiment responses.

Other research has been conducted on opinion and sentiment posted on Twitter. Yang et al [Yang et al., 2007] used web-blogs to build corpora for sentiment analysis and used emotion icons allocated to blog posts as indicators of users' mood. The authors applied Support Vector Machine *(SVM)* and Conditional Random Field *(CRF)* learners to categorise sentiments at the sentence level and then examined numerous approaches to regulate the overall sentiment of the document. The sentiment of the latest sentence of the document is then considered as the sentiment at the document level [Read, 2005]. Tumasjan et al [Tumasjan et al., 2010] used Linguistic Inquiry and Word Count *(LIWC)* text analysis software to conduct a content analysis of the German federal election of 2009. They examined whether Twitter can be used as a medium for political discussion and whether on-line tweets can be mapped to political sentiment in the real world. Their investigation confirmed that tweets could be used as a platform for political discussion.

2.4 Topic Detection and Tracking on Twitter

It has been established that not all tweets are event or topic related. Twitter users are known to tweet for different reasons ranging from expression of personal mood, opinion on on-going issues/topics [Liu, 2012] to information dissemination [Lerman and Ghosh, 2010] in real-time [Adedoyin-Olowe et al., 2013, Agarwal et al., 2012, Chakrabarti and Punera, 2011, Kwak et al., 2010, Weng and Lee, 2011]. Research on event detection is an old topic [Allan et al., 1998]. Event can be referred to as a single occurrence of interest happening at a specific period and location [Sayyadi et al., 2009]. Tweets relating



:)	happy	:/	unsure	>:O	upset
:(	sad	:'(	cry	:v	pacman
:P	tongue	3:)	devil	:3	curly lips
:D	grin	O:)	angel	:|]	robot
:O	gasp	:*	kiss	:putnam:	
;)	wink	<3	heart	(^^^)	shark
B)	glasses	^_^	kiki	<(")	penguin
B|	sunglasses	-_-	squint	:42:	42
>:(	grumpy	o.O	confused	(y)	thumb

Figure 2.3: Common Social Media Emotion Icons.

to an event always result in the surge in the use of certain keywords during the occurrence of the related event in the real world [Kleinberg, 2003].

Topics/events detection and tracking has been widely researched since the last decade. Diverse TDT methods are being used to detect relevant events and news topics embedded in on-line tweets. Events related tweets, (also refer to as event tweets), ranges from sports [Guzman and Poblete, 2013, van Oorschot et al., 2012], politics [Ausserhofer and Maireder, 2013], stock market [McCreadie et al., 2013]. N-grams method effectively capture intricate combination of tweets in real-life topics of diverse composite and time scale by recognising trend in the topics [Aiello et al., 2013]. Another TDT method as proposed by [Agarwal et al., 2012] was applied to tweets to analyse real-life events and occurrences such as sparsely reported events. Their generic architecture for transforming a tweet-stream into event-objects uses locality sensitive hashing, classification, boosting, information extraction and clustering. They affirmed event correlation as two-step progression, with the first consisting the raw message level and the second through semantic analysis of events. Tweets relating to real-world events and non-event were differentiated using Real World-Event *(RW-Event)* classifier was proposed by [Becker et al., 2011]. TwitterMonitor method was proposed in the work of [Mathioudakis and Koudas, 2010] and used to monitor topic trend (emerging topics) on Twitter in real-time. The method offered significant analytics that produces a precise description of each topic. the authors of [McCreadie et al., 2013] built a system that detects scalable distributed event as well as characterises emerging trends. They use lexical key splitting approach to spread the event detection procedure through various machines, while evading divide-and-conquer approaches that divides and develops the stream as a series of sub-sets. The study of [Naaman et al., 2011] on event detection made two contributions to the interpretation of emerging temporal trends. These includes the development of a taxonomy of the trends present in the data and the identification of significant dimensions of trend categorisation, as well as the key distinctive structures of trends that can be resultant from their related tweets. The authors of [Watanabe et al., 2011] proposed an automatic geotagging method that uses geo-location information and timestamps of tweets to detect local events happening between the vicinity of users' current location in real-time using on-line tweets.

Incremental on-line clustering and filtering framework is used to distinguish between messages about real-life events and no-events [Becker et al., 2012]. The framework clusters subsequent tweets-based message similarity with existing clusters. On the other hand, graph-based approaches can detect keyword clusters in tweets based on their pairwise comparison [Inouye and Kalita, 2011,Aiello et al., 2013]. This can be a term unison graph with nodes clustered and the use of community detection algorithm based on betweenness centrality [Sayyadi et al., 2009]. Graph-based methods can also be applied to evaluate the effectiveness of topic extraction from tweets [Meng et al., 2012]. Jackoway et al [Jackoway et al., 2011] used a clustering technique to detect events using a text classifier. The work of [Ritter et al., 2012] proposed a scalable method of mining and classifying events from ranked messages in an open-domain text genre with unidentified categories. Their method was based on latent variable models and detects events types that match the data and subsequently categorise the collective events without annotated samples.

Peculiar controversial events capable of triggering public discussion on Twitter are discovered by machine learning *direct model*, *two-step pipeline model* and *two-step blended model* [Popescu and Pennacchiotti, 2010]. Topically related message clusters can be applied to spot real-world events and non-events messages on Twitter [Becker et al., 2011]. Similarly EDCoW (Event Detection with Clustering of Wavelet-based Signals) is able to cluster words to form events with a modularity-based graph partitioning method [Weng and Lee, 2011]. Real-time and dynamic events such as sports on Twitter network can be encapsulated using SUMMarising Hidden Markov Model *SUMMHMM* [Chakrabarti and Punera, 2011]. The model is used to formalise the issue of summarising events tweets and proffering solution based on gaining knowledge from the fundamental concealed formal demonstration of the occurrence via Hidden Markov Models. Page Rank algorithm is used to retrieve tweets of users with authority on the network [Cataldi et al., 2010]. The retrieved tweets are used to create a navigable topic graph that connects the emerging topics under user-defined time slot. The authors of [Walther and Kaisser, 2013] proposed an algorithm for geo-spatial event detection. They evaluated the consequential spatio-temporal clusters of tweets using a Machine Learning element to identify real-life and non-real-life events.

In [Corney et al., 2014] n-grams and $df\text{-}idf_t$ was used to group together terms that appear in the same tweets with a standard hierarchical clustering. They identified term clusters whose similarities are high as a representation of the same topic and merged clusters to the point where each cluster is assumed to signify a distinct topic. They presented a more comprehensive detail of their algorithm in [Aiello et al., 2013] by identifying real world topic in the 2012 US Presidential Elections, the US Super Tuesday 2012 and the English FA Cup 2012. TwitterMonitor system was built to detect trend of emerging topics and new story on Twitter in real-time [Mathioudakis and Koudas, 2010]. Detection of new story emerging topic on Twitter is discussed in Section 2.4.1 and Section 2.4.3.

2.4.1 First Story Detection on Twitter

First Story Detection *FSD* (also refer to as ‘new event detection’) is a subtask within TDT [Allan, 2002a]. First story detection is when a text comprises new content that has not been accessed previously [Zhang et al., 2011]. The essence of FSD is to identify the initial source broadcasting of specific story. Out-of-date methods for FSD denotes text as vectors using space and denoting the rate of specific term in a text [Petrović et al., 2010]. Two major challenges of detecting first story on Twitter is that of data volume and noise. It is relatively more complex to detect first story on the network when compared to traditional news media first story broadcast. Detecting first story requires the application of algorithms capable of handling Twitter streaming data.

FSD structures are created on the basis of documents as vectors within a duration using term frequencies [Allan, 2002b, Yang and Honavar, 1998]. Distance measurement is used to detect first story, this is obtained by comparing new documents to their nearest neighbour by measuring their distance gap. Documents with distance that exceed a predefined maximum value are considered as first story. This method collects all documents term frequencies in memory and detect the nearest neighbour for in-coming documents [Indyk and Motwani, 1998]. Recently identified issues associated with streaming data include topic modelling on streaming content collections [Yao et al., 2009], stream-based machine interpretation [Levenberg and Osborne, 2009] and estimating kernel matrices of data streams [Shi et al., 2009].

An improved Locality Sensitive Hashing (LSH) was proposed to [Petrović et al., 2010] to search for nearest neighbour enhancement that satisfies the data stream mining prerequisites using constant size buckets. In [Osborne et al., 2012] a method that merges Twitter and Wikipedia in order to enhance event detection was presented. They explored the latency between the two streams and discovered that Twitter is more up-to-date in real-life events posting. While new story detection in real-life is achieved by simply comparing existing stories with in-coming ones to ascertain newness, detecting a new story on Twitter comes with computational challenges.

2.4.2 Detecting Breaking News

While first story detection comprise of new content that has not been accessed previously in a text, breaking news entails newly acknowledged information about an event that is currently taking place or emerging. Breaking news often comprises incomplete information that is subject to updates as the news unfolds. Twitter is known for information dissemination in real-time, this makes it a renowned network for breaking news. Even when there are no traditional newsagents at the scene of an impromptu event, there will always be Twitter subscribers present at the scene and ready to post the event on Twitter in real-time before newsagents arrive at the scene.

The authors of [Phuvipadawat and Murata, 2010] proposed a method of collecting, grouping, ranking and tracking breaking news in Twitter. The authors built a framework named 'Hotstream' to enable users discover breaking news from Twitter time-line. The work of [Petrovic et al., 2013] established that, although Twitter cannot be said to post breaking news on-line faster than newswire, the network covers a greater part of hyper-local news stories that is overlooked by newswire. In [Meyer et al., 2011] a technique that collects data, detects breaking news topics, and shows outcomes in a geo-temporal visualisation was presented.

2.4.3 Detecting Emerging Topics on Twitter

There are currently numerous methods for viewing emerging topics on Twitter such as *Twitscoop, Trendistic, Twopular* and *Trend on Twitter.com* [Elvers and Srinivasan, 2011]. News media and businesses are always eager to detect hot news on the network as quickly as possible. Emerging topic on Twitter can be captured by significant hashtag keywords that are relevant to a real-life occurrence [Chang, 2010]. These topics often result into trending topics on Twitter and can be categorised and characterised based on location [Naaman et al., 2011]. Localisation of emerging trends enables the identification of local events.

Furthermore, considering the social relationship in the user network can serve as an authenticity measurement for every analysed tweet. Emerging stories can be detected with the dictionary learning [Kasiviswanathan et al., 2011] using two-stage approach based on detection and clustering of novel user-generated content. Using key-phrases based on statistical language models for matching the comparative frequency of phrase occurrence at two time periods is found to be more efficient in detecting emerging topics more than high frequency of terms [Tomokiyo and Hurst, 2003]. The work in [Cheong and Lee, 2009] considered four trending topics and two control terms, as well as a subset of tweets related with each of them. They pass remarks on structures like time-based frequency for each term, and the group of users and types of devices used to post the related tweets. The work of [Chen et al., 2013] proposed semi-supervised learners to develop real-time framework targeted at detecting hot emerging and evolving topics relating to specific organisations in Twitter before they become hot topics. Their experiments revealed the relevance of features such as the rates of accumulating number of tweets, re-tweet as well as the whole collective influence of tweets in contributing to good performance of hot emerging topic detection on the network. On the other hand, [Takahashi et al., 2011] applied probability model captured the number of "mentions" in each tweet and the degree of users posting the mentions. They assembled the mention model with the SDNML change-point detection algorithm and the Kleinberg's burst detection model to determine an emerging topic.

2.5 Summary

Having reviewed the different techniques currently used for TDT on Twitter, as listed in Tables 2.1, 2.2 and 2.3, it was observed that most of the techniques proposed in the literature were limited to extracting specific types of scenarios in events. Scenarios such as breaking news using 'Hotstream' [Phuvipadawat and Murata, 2010], emerging news using dictionary learning [Kasiviswanathan et al., 2011], event tweets using Real World-Event *(RW-Event)* classifier [Becker et al., 2011], trending news using localization [Naaman et al., 2011] and first story using Locality Sensitive Hashing (LSH) [Petrović et al., 2010] were used on Twitter data. Unlike the reviewed techniques presented in this Chapter:

- We adopted **Association Rule Mining** to build *TRCM* system which, is capable of classifying all the types of events detected on Twitter at **once** with the same method. *TRCM* and all its rules are defined and explained in details in Chapter 4.
- Experiments conducted in Chapter 5 & Chapter 6 classify all hashtag keywords detected under the four *TRCM* rules.
- *TRCM* revealed that breaking news often evolve into *emerging rule* with high frequency of its related hashtag keywords(s) in the specific time window presented as **strong** rule.

Table 2.1: Twitter Analytics Table

Approach	Tools	Experiments	Authors/dates
Opinion Mining	Analysis	Emoticon Analysis	Kouloumpis et al (2011)
Sentiment Analysis	Psychometric instrument	Modelling Public Mood and Emotion	Bollen et al (2011)
Sentiment Analysis	Granger causality analysis & Fuzzy Neural Network	Predicting the stock market through Twitter mood	Kouloumpis et al (2011)
Opinion Mining	SVM, Naive Bayes, Maximum Entropy, Artificial Neural Networks	Exploitation of influence factor to predict the outcome of the US election in 2012	Anjaria et al (2014)
Sentiment Analysis	Visual illustration & Metrics	understanding sentiment expressed through tweets of in political debate in 2008	Diakopoulos and Shamma (2010)
Sentiment Analysis	SVM, latent semantic analysis	Prediction of political orientations of Twitter users based on the content and system of their political messages in the US midterm elections in 2010	Conover et al (2011)
Sentiment Analysis	SVM, conditional random field (CRF)	Use of web-log in building a corpora for sentiment analysis and use of emotional icons allocated to blog posts as indication of users mood	Yang et al (2007)
Sentiment Analysis	LIWC	conducting content analysis of the German federal elections of 2009	Tumasjan et al (2010)
Event Detection	Locality sensitive hashing, classification, boosting, information extraction, clustering	Analysis of sparsely reported real-life events	Agarwal et al (2012)
Event Detection	Real World Event RW-Event Classifier	Differentiate between real world event and non-event tweets	Becker et al (2011)

TABLE 2.2: Twitter Analytics Table (Continues)

Approach	Tools	Experiments	Authors/dates
Event Detection	Lexical key splitting approach	Detection of scalable distributed events and characterises emerging trends	McCreadie (2013)
Event Detection	Taxonomy	Ascertaining the significance of scope for categorising trend on Twitter	Naaman et al (2011)
Event Detection	Automatic geo-tagging	Detection of local events within user's current location	Watanbe et al (2011)
Topic Extraction	Graph-based	Evaluation of of the effectiveness of topic extraction from tweets	Meng et al (2012)
Event Detection	Text classifier, clustering	Identification of Live News Events using Twitter	Jackoway et al (2011)
Event Detection	Incremental on-line clustering	Distinguishing between event and non-event tweets	Becker et al (2012)
Event Detection	Latent variable models	Detection of events types using ranking messages in open-domain text genre with unidentified categories	Ritter et al (2012)
Topic Detection	Direct model, two-step pipeline model, two-step blended model	Detection of controversial topic on Twitter	Popescu & Pennacchiotti (2010)
Topic Detection	Message clustering	spotting real-life events and non-events tweets	Becker et al (2011)
Event Detection	EDCoW (Event Detection with Clustering of Wavelet-based Signals)	Clustering of word to form events	Weng & Lee (2011)

Table 2.3: Twitter Analytics Table (Continues)

Approach	Tools	Experiments	Authors/dates
Event Detection	Summarising Hidden Markov Models (SUMMHMM)	Summarising sports event tweets	Chakrabarti & Punera (2011)
Information Retrieval	Page ranking	retrieval of tweets of influential users in Twitter	Cataldi et al (2010)
Event Detection	N-grams,$df - idf_t$, hierarchical clustering	detecting event highlights in FA Cup game	Corney et al (2014)
Event Detection	N-grams	Topic/event detection in the FA Cup finals 2012.2013, US Presidential elections 2012 and the Super Tuesday 2012	Aiello et al (2013)
Event Detection	TwitterMonitor	Monitor emerging topics on Twitter in real-time	Mathiousdakis & Koudas (2010)
First Story Detection	Clustering	Comparison of report speed between Twitter and Newswire	Petrovic at al (2010)
First Story Detection	Latency	Comparison of report speed between Twitter and Wikipedia	Osborne et al (2012)
Breaking New Detection	Hotstream	Discovery of breaking news from Twitter users' timeline	Phuvipadawat & Murata (2010)
Breaking News Detection	Geo-temporal visualisation	Automatic identification of breaking news events in near real-time	Meyer et al (2011)
Emerging Story Detection	Dictionary learning, clustering	Using two-stage approach based on detection and clustering of novel user-generated content	Kasiviswanathan et al (2011)
Emerging topic Detection	visualisation	Collective intelligence retrieval with activated knowledge-based decision making	Cheong & Lee (2009)

- *Unexpected rule* of *TRCM* represents twist in an on-going event in the real world. This type of twist reveals a new aspect of the on-going topic/event/news that is worth considering. An example of such occurrence is given in Chapter 4 of this thesis. The authors of [Mathioudakis and Koudas, 2010] applied TwitterMonitor system to detect trending topics on Twitter network, their algorithm considered *'bursty' keywords* that suddenly appear in tweets at a rapid rate and group them into clusters to identify trending topic. They applied an algorithm termed "QueueBurst" that reads streaming tweets only once and declare any 'bursty' keyword detected. By means of this, the chances of misrepresentation of trending keyword cannot be completely ruled out due to spurious bursts and spam posts. However, there was no description of how the system is tuned to by-pass these noise.

- *TRCM* consider only hashtag keywords as **items** in a **transaction** represented by tweets in a transactional database of Association Rule Mining.

Considering the fact that hashtag label is meant to describe tweets' contents, *TRCM* is trained to extract only hashtags present in tweets and to detect frequent itemsets (hashtags) that co-occur at two consecutive time period (t+t+1). The frequent hashtags at $t+1$ are matched with those at t to detect rule similarity. Rule similarity found to be equal or greater than the user-defined threshold are classified under the appropriate *TRCM* and mapped to related real-life topic/event. The methodology was able to cope with most of the Twitter challenges discussed in Section 1.4 such as *noise* & data volume by extracting only tweets with hashtags that best represents the specific topic/event by using official hashtag keywords of such topic/event. The research methodology also considered tweets written in English in order to address the issue related to *multilingual tweets*. *ARM* is a holistic framework, it accentuates the significance of Twitter data and the inter-dependence of hashtag keywords. To the best of our knowledge, no work has applied *ARM* to hashtags to TDT on Twitter. The research reported in this thesis not only detects real-life topics from tweets but also tracks topics as they evolve on Twitter and in reality. The method focuses on tweets with hashtag labels that represent the targeted topic/event. The next chapter gives an overview of *ARM*, the data mining technique used to generate rules in this research.

Chapter 3

An Overview of Association Rule Mining

3.1 Introduction

In order to be able to describe *TRCM* in details, it is necessary to discuss Association Rule Mining *(ARM)* and its task of extracting interesting relationships between items in large datasets. This is extracted in form of frequency and Association Rules *(ARs)*. While frequent itemsets are those items that co-occur frequently, as *ARs* learns the strong associations that occur between two items. Whereas association rule is mostly used in retail business, in the experiments conducted in this thesis, we apply *ARs* to tweets' hashtags of specific tweets relating to specified real events to discover hashtags that best describe the targeted event in real-life. This chapter gives a comprehensive analysis of *ARM*. It also gives a comparison between the two main tools of *ARM* namely: the

Apriori and ***FP-Tree*** methods. We then justify our adoption of the Apriori method in the research.

3.2 Association Rule Mining Approach

ARM is a data mining technique used for mining significant associations rules common to different collections of items in data repositories such as transactional and relational databases [Agrawal et al., 1993, Liu et al., 2009]. It extracts interesting recurrent representation, associations or links, between different arrays of items within transactional databases (market basket), relational databases (eg. personal details), or any other information repositories [Liu et al., 2009] in the form of rules. *ARM* also discovers and reveals remarkable associations embedded in huge data sets which may include hidden information that can be useful for decision making [Jain et al., 2012]. The technique tends to reveal every probable association that satisfies definite boundaries using the defined minimum support and confidence [Ale and Rossi, 2000]. *ARM* is mostly used for Market Basket Analysis *(MBA)* to detect frequency of specific items within the dataset. It evaluates the frequent antecedent/consequent patterns by using the support and confidence measures to detect significant relationships [Brin et al., 1997b] that satisfies the user-defined support and confidence thresholds. For instance, *ARM* enables business owners to understand their customer purchasing behaviour. It is used to ascertain items that are purchased together, for example $bread, milk \Rightarrow egg$. This shows that customers who buys bread and milk also buys egg. *ARM* enables stores to discover which items sell faster together and those that are not frequently sold. This analysis assist businesses when making important business decisions. Items purchased together can be placed within close proximity and those that sells less frequently together can be put on offer to attract increased sales.

The rule form for *ARM* can be demonstrated as follows:

$$Antecedent \rightarrow Consequent(user-defined)[support,\ confidence]$$

$$Examples: buys(y,"dress") \rightarrow buys(y,"shoes")[0.5\%.60\%]$$

$$gender(y,"female") \wedge income(y,"50000-55000") \rightarrow buys(y,"house")[1\%,75\%]$$

3.3 Measuring Rule Interestingness

In classification the quality of the ruleset is vital, it is expected that the summation of the rules would determine the performance effectiveness of the classifier rather than any individual rule [GENG and HAMILTON, 2006]. However, this is not the case in *ARM*. In *ARM*, emphasis is placed on the quality of each rule. In order to differentiate among rules, it is necessary to measure their quality. This measurement is termed *rule interestingness measure*. The interestingness measure in *ARM* can simply be described using four numerical values which can be used to ascertain any rule describe as computation of the *IF, THEN* statement first (as shown in Table 3.1), then by using the venn diagram format as described in [Bramer et al., 2007] and shown in Table 3.1).

TABLE 3.1: IF, THEN of *ARM*

$N_{Conditional}$	Number of instances matching Left
$N_{Consequent}$	Number of instances matching Right
N_{Both}	Number of instances matching both the Left and the Right
N_{Total}	Total number of instances in the dataset

Using the venn diagram format as presented in Figure 3.1, the outer box of the venn is labelled as a container for all N_{Total} instances in the dataset under

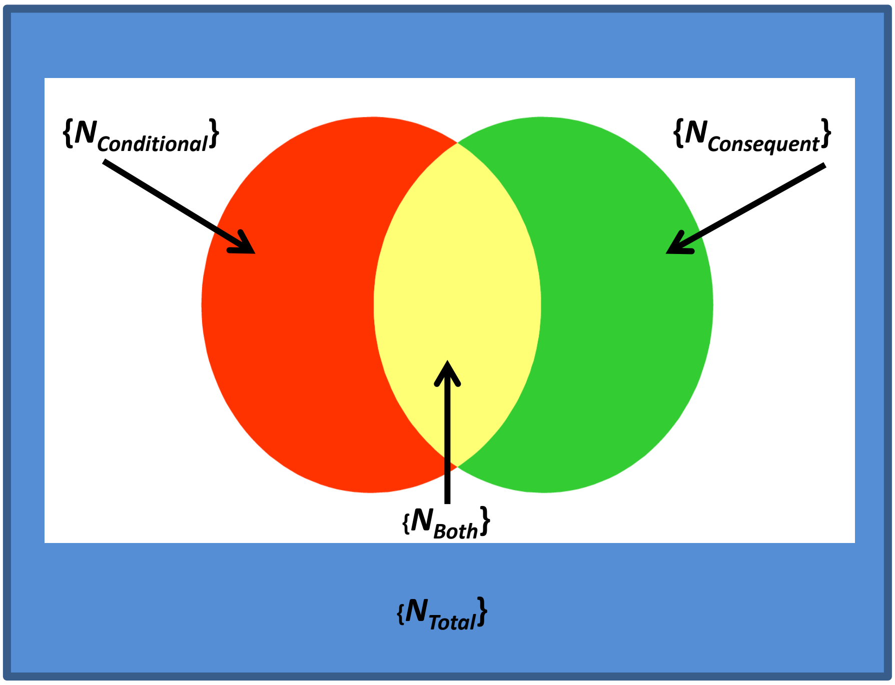


FIGURE 3.1: Matching Instances in the Left, Right and BOTH Left and Right.

review. The left and the right hand circles are labelled as container for the $N_{Conditional}$ instances that match the *left* and the $N_{Consequent}$ as those that match the *right* part of the circle. The middle where the circles interlock contains the N_{Both} instances which match both the *left* and *right*. In this research, hashtags describes items while tweets signifies transactions. All the rules are explained in details in Chapter 4.

Confidence (Analytical Correctness, Dependability)

$N_{Both}/N_{Conditional}$

This signifies the proportion of the right-hand sides as predicted by the rule that are correctly predicted.

Support

N_{Both}/N_{Total}

This signifies the proportion of the training set accurately predicted by the rule.

Completeness

$N_{Both}/N_{Consequent}$

This describes the proportion of the matching right-hand sides that are accurately predicted by the rule.

Most interestingness measures can be calculated using the venn diagram in Figure 3.1 despite its rudimentary illustration. The high volume of possible rules incresases the chances of irrelevant rules during the dataset exploration [Bramer, 2013]. To search and detect pattern in the dataset, **pruning** becomes necessary in addition to the use of frequency measures. Pruning is completed using the interestingness and the potential measures:

- interestingness: This determines whether the degree of interestingness of any identified pattern is adequate for extraction.
- potential: It shows whether the identified pattern reveals a promising interesting knowledge.

Interestingness of detected patterns is relative, a likely pattern may not be interesting [McGarry, 2005]. This can be further measured using the accuracy and frequency of the pattern collectively with the contextual knowledge.

3.4 Support Measures in Association Rule Mining

Support (S) of an itemset I is the proportion of transactions in the database that is matched by I [Bayardo Jr and Agrawal, 1999]. This means that an itemset I matches a transaction T which is part of the whole itemset, where S

is a subset of T [Bramer, 2013,Hipp et al., 2000]. The frequency with which the items in I occur collectively in the database is considered. Where $support(S) = count(S)/n$, given n is the number of transactions in the database. The rule $X \Rightarrow Y$ supports if the % support of transactions in T contains $X \cup Y$. Support can also mean a *fractional support* which means the proportion of transactions that supports X in T. *Support* can be summarise as follows:
Let $K = \{k_1, k_2, k_3..., k_n\}$ be a set of items, let D (the database), be a set of transactions T with each transaction representing the set of items [Srikant and Agrawal, 1996]. T is said to support an item x if x occurs in T, while T supports a subset of items X. $X \Rightarrow Y$ holds if *support* s in $s\%$ of the transactions in D that supports X also support Y. This implies that T supports a subset of items X.
Rules that have support equal or greater than a user-defined support is said to satisfy the minimum support. *Apriori* algorithm allows for multiple setting of minimum support threshold without affecting the process of frequent items and rules extraction. Support can be calculated using equation 3.1. An example to further illustrate support is given in Table 3.2.

$$Support(A \Rightarrow B) = P(A \cup B) \tag{3.1}$$

Table 3.3 shows that the frequency of itemsets in Transaction T. In the transactions {a} has frequency of 75%, {b}, 62.5%, { c}, 87.5% and {a, b, c}, 50%.

In the experiments conducted in this thesis, hashtags is handled as items and the tweets is treated as the transactions in the database (as shown in Table 3.4).

TABLE 3.2: Sample of transactional Database Showing Frequency of Itemsets

Transaction ID	Transactions
1	$\{a, b, c\}$
2	{a, b, c, d, e}
3	{a, b}
4	{c, d, e}
5	{a, c}
6	{a, b, c, d}
7	{c, d, e}
8	{a, b, c, e}

TABLE 3.3: Support of Itemsets in Transactional Database

Transaction ID	Transactions
{a}	75%
{b}	62.5%
{ c}	87.5%
{a, b, c}	50%

3.5 Confidence Measures in Association Rule Mining

The rule $X \Rightarrow Y$ suffice with *confidence (c)* of $c\%$ of the transactions that includes X also includes Y [Agrawal et al., 1993]. Confidence is used to create rules from the frequent itemsets by extracting only rules with c equal to or

Table 3.4: The Tweet Matrix

Tweet 1	#datamining	#bigdata	#sql	#KDD
Tweet 2	#ecommerce	#ISMB	#datamining	
Tweet 3	#bigdata	#facebook	#data mining	#analytics
Tweet 4	#analytics	#privacy	#datamining	
Tweet 5	#datamining	#KDD	#bigdata	

greater than the user-defined minimum confidence (min_conf) holds as presented in Tables 3.5 and 3.6.

$$confidence(A \Rightarrow B = P(B|A) = \frac{support\_count(A \cup B)}{support\_count(A)} \tag{3.2}$$

Table 3.5: Confidence of Itemsets in Transactional Database

Transaction ID	Transactions
1	Dress, Shoe, Bag, Belt
2	Dress, Shoe, Bag, Belt, Necklace, Hat
3	Dress, Bag
4	Dress, Shoe, Belt

$Shoe \Rightarrow Bag$

$Support - 50\%(2/4)$

$Confidence - 66.67\%(2/3)$

Table 3.6: Confidence of Itemsets in Transactional Database

Transaction ID	Dress	Shoe	Bag	Belt	Necklace	Hat	Sandal
1	1	1	1	1	0	0	0
2	1	1	1	1	1	0	1
3	1	0	1	0	0	0	0
4	1	1	0	1	0	0	0

3.5.1 Discovering large itemsets

Discovering large itemsets in a database requires the algorithm applied to make several passes over the dataset [Agrawal et al., 1994]

3.6 Lift in Association Rule Mining

The major concern of support and confidence is that of establishing a valid means of deciding the suitable values for *min_sup* and *min_conf*. Setting *min_sup* that is too high will result in missing important rules, while setting it too low will generate too many rules, some of which might be irrelevant [Liu et al., 1999]. Some rules having uncommon itemsets might be of interest in some situations but the concept of correlation are not being captured. A rule $A \Rightarrow B$ that satisfies both the *min_sup* and *min_conf* constraint may not have any correlation between A and B, which means that $support\ (A) \times support(B) = support(A \cup B)$.
Lift choose rules that have high score of importance and interestingness [GENG and HAMILTON, 2006]. It denote the relation and the difference between the support and if the support would have occurred if A and B are autonomous.

It tends to detect rules with strong correlations between A and B as shown in equation 6.1.

$$lift(A \Rightarrow B) = \frac{confidence(A \Rightarrow B)}{confidence(\emptyset \Rightarrow B)} = \frac{support(A \Rightarrow B)}{support(A) \times support(B)} \tag{3.3}$$

Lift is considered in the visualisation conducted on the qualitative case study in Chapter 6 of this thesis. Rules with a minimum lift of 1.92 were selected for visualisation. This further enhances the detection of hashtags that falls under the strong association rules categories.

3.7 The *Apriori* Approach to Association Rule Mining

Apriori method is a common algorithm for learning *ARs* for boolean associations [Srikant et al., 1997, Joshi and Sodhi, 2014]. Based on **prior knowledge** of frequent itemset properties, *Apriori* uses an iterative method named *level-wise* search to detect frequent itemsets and strong *ARs* [Jiawei Han, 2011] as presented in Fig. 3.2.

This is achieved by generating a set of all probable combination of items and subsequently computing the support for the itemsets. The downward closure property of frequent patterns *(*k-itemset) implies that any subset of a frequent itemset must be frequent [Zaki and Hsiao, 2002] *(*k-1) as described as follows:

- If a transaction containing {Shoe, bag, belt} is also containing {Shoe, belt}; {Shoe, bag}; {bag, belt}

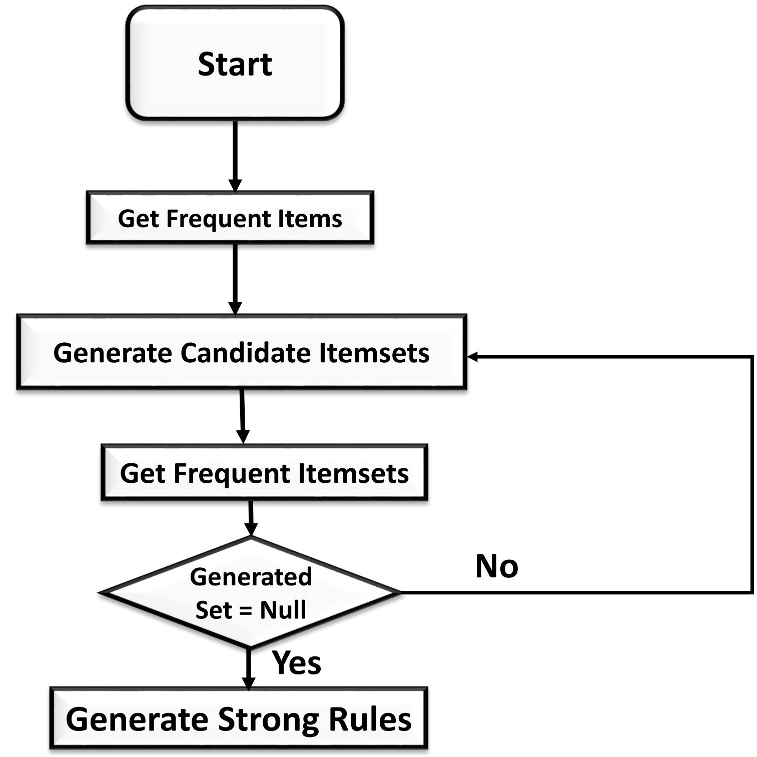


FIGURE 3.2: How *Apriori* Works

- $\{Shoe, bag, belt\}\ isfrequent \rightarrow \{Shoe, belt\}$ MUST also be frequent. Any superset of an infrequent itemset are also infrequent and are eradicated from the rule generation.

3.7.1 The Algorithm Components of *Apriori*

Since the purpose of data mining techniques (including *Apriori*) is to solve specific task, it is imperative to define identified components of the technique. We based our explanations on the work of [Hand et al., 2001] and the components include:

1. The use of appropriate technique to interpret/address the task; whether classification, clustering, regression or visualisation.

2. Verification of the model structure adopted to fit the data. The structure encompasses the margins within which learning is effected.

3. The Score Function *(SF)* used to evaluate the quality of the fitted models based on the observed data (for example, classification error). The *SF* can either be maximized or minimized when parameters are fitted to the models/patterns. The *SF* is vital to for learning and generalisation of the models. For *Apriori*, the *SF* used is **accuracy**. Given that the *ARM* is only a component in the *TRCM* system, other matrix were used to assess the performance of the technique for topic/event detection and tracking. For *ARM*, method like *Apriori* rule with favourable interestingness measures can be used as score function. More details about this matrix is given in Chapter 4.

4. *The search and optimization method* is applied to search the parameters and structures such as computational processes and algorithms used to identify the maximum/minimum of the score function for specific models/patterns. The concerns arising from this identification include the computational methods employed to enhance the *SF*, for example, search-related parameters such as the maximum number of iterations or convergence depiction for an iterative algorithm. For a pattern of a single fixed system such as the *kth* order polynomial function of the data, the search is carried out in the parameter space to enhance the *SF* comparative to the fixed structural form as it is the case of the pattern of the data used in this research. In *ARs*, the search is done in accordance to the employed technique, for example, *Apriori* apply the greedy search to find frequent itemsets.

5. The *Data Management Technique* employed for storing, indexing and mining data. Accessing large datasets from secondary storage may affect the efficiency of the algorithms, therefore the location of the data and the methods of accessing it are vital. The five components of *Apriori* discussed are presented in Table. 3.7.

Table 3.7: The Algorithm Components of *Apriori*

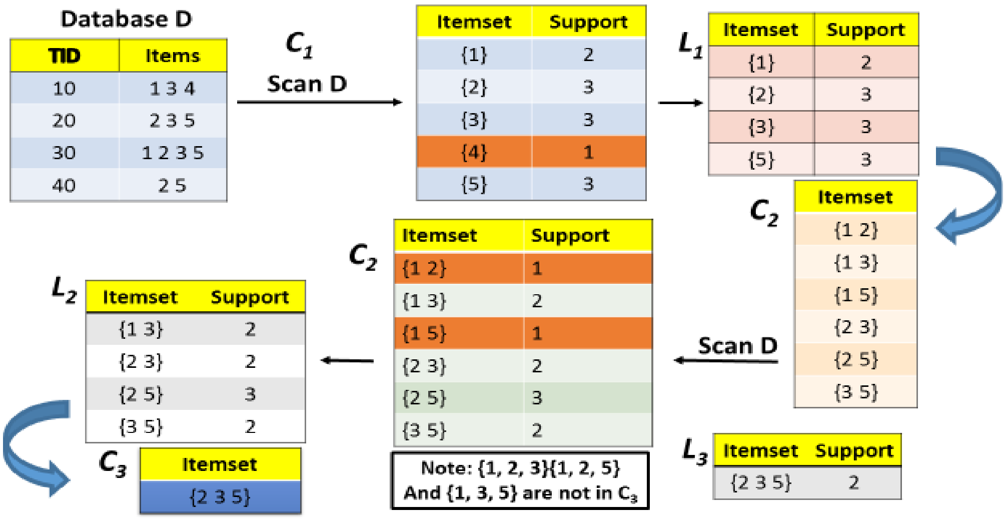
Component	Definition
Task	Rule Pattern Discover
Structure	Association
Score Function	Support/Accuracy
Search Method	Breadth-Fast with Pruning
Data Management Technique	Linear Scans

The explanations of how **Apriori property** is used to reduce the search space during frequent itemsets generation is based on the work of [Jiawei Han, 2011]. *Apriori property* generate frequent itemsets of length 1. If an itemset I does not satisfy the min_sup threshold, then I is termed infrequent $P(I) < min\_sup$. However, if a item A is introduced to the itemset I, then $I \cup A$ cannot be more frequent than I. This means $P(I \cup A) < min\_sup$. This process is referred to as **antimonotonicity** which interprets that if a set fails to satisfy the min_sup threshold, then all of its subsets will also not satisfy the min_sup. To use this algorithm, *Apriori property* applies a two-step process containing of the **join** and **prune** process. To understand this, let us consider how L_{k-1} can be employed to find L_k for $k \geq 2$, L_k is frequent itemset of k.

- **The join level**: To identify L_k, a set of candidate k-itemsets is created by joining L_{k-1} to itself. The set of candidates indicates C_k. Let l_1 and

l_2 be itemsets in L_{k-1}. The representation $l_i[j]$ points to the *jth* item in l_i ($l_1[k-2...]$ points to the second up the last item of l_1. For operational application, *Apriori* concedes that items in the transaction are sorted in lexicographic format. For the $(k-1)$-itemset, l_i, this denotes that the items are categorised in a way that $l_i[1] < l_i[2] <\ldots$ l_i[k - 1].

- **The prune level**: C_k is a superset of L_k, which implies that its members may or may not be frequent, whereas all of the frequent k-itemsets are contained in C_k. L_k is decided in a database by scanning the count of each candidate in C_k (for example, all candidates with a count greater than the *min_sup* count are frequent and therefore related to L_k).



TID	Items
10	1 3 4
20	2 3 5
30	1 2 3 5
40	2 5

Itemset	Support
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

Itemset	Support
{1}	2
{2}	3
{3}	3
{5}	3

Itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

Itemset	Support
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

Itemset	Support
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

Itemset
{2 3 5}

Note: {1, 2, 3}{1, 2, 5} And {1, 3, 5} are not in C_3

Itemset	Support
{2 3 5}	2

Figure 3.3: Diagram Showing Apriori Process.

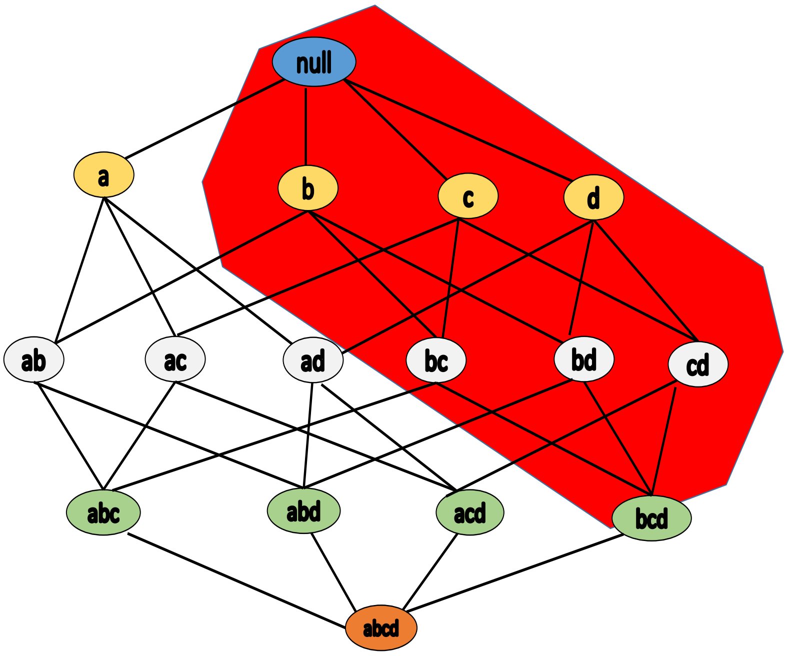


FIGURE 3.4: Diagram Showing Frequent Itemsets Generation Using Apriori. if {b,c,d}is frequent, then **all subsets** of {b,c,d} are also frequent

3.8 Apriori Algorithm and Itemset Generation

Consider a database, D in Figure 3.3, the minimum support threshold is the lowest support of itemsets to be extracted. The output of the algorithm is frequent itemsets that satisfies the pre-defined minimum support threshold as showned in Figure 3.4 and it is achieved as follows:

1. Scan D for count of each candidate.
2. Compare candidate support count with minimum support count. The set of frequent-itemsets L_1 consist of the candidate 1-itemsets satisfying minimum support. In the first iteration of the algorithm, each item is a member of the set of candidates.

3. To discover the set of frequent 2-itemsets, L_2, the algorithm employ L_1 *Join* L_1 to generate a candidate set of 2-itemsets, C_2.

4. The transactions in D are then scanned and the support count for each candidate itemset in C_2 is accrued. The set of frequent 2-itemsets, L_1, is then ascertained, consisting of those candidate 2-itemsets in C_2 having minimum support.

5. The generation of the set of candidate 3-itemests, C_3, includes the use of the *Apriori* Property.

6. To find C_3, L_2 *Join* L_2 is computed.

7. itemsets containing subsets of length k frequent itemsets are **pruned** as candidate.

8. **Count the support** of each candidate by scanning the database.

9. Eliminate candidates that are not frequent, retaining only those that are frequent. As presented in Fig. 3.4 all the sub-sets of frequent itemsets will also be frequent.

Apriori algorithm is presented in Algorithm 1

Algorithm 1 Apriori

Input:

D: transaction database;

Min_sup: the minimum support threshold

Output: frequent itemsets

Description:

```
$L_1$= find_frequent_1-itemsets(DB);
for (k=2; $L_{k-1}$is non-empty; $k++$) {
$C_k$= Apriori_gen($L_{k-1}$);
for each transaction $t \in DB$ {    //scan DB for counts
$C_t$ = subset($C_k, t$);   //get the subsets of $t$ that are candidates
for each candidate $c \in C_t$
$c.count++$;
}
$L_k = \{c \in C_k | c.count \geq min\_sup\}$
}
return $L = \bigcup_k L_k$;
Procedure Apriori gen($L_{k-1}$: frequent($k-1$)-itemsets)
```

3.9 Improving the *Apriori* Using Hash-based Technique

Many variations of improving the efficiency of the original *Apriori* approach have been proposed [Singh et al., 2013, Li et al., 2012]. One of these variations is the **Hash-based technique** [Park et al., 1997]. This technique is capable of reducing the size of the candidate k-itemsets, C_k, for $k > 1$. During the process of scanning each transaction in the database to generate the frequent

1-itemsets in C_1, all the 2-itemsets for each transaction can be generated. The items can then be mapped into different container of a hash table structure, while increasing the corresponding container counts as shown in Fig. 3.5. The process is like hashing itemsets into well-matched containers. A 2- itemsets whose corresponding container count in the hash table is less than the support threshold are infrequent and should be eliminated from the candidate set. Such a hash-based technique can significantly condense the number of the candidate k-itemsets scanned (especially when $k = 2$). The algorithm for Hash-based method is presented in Algorithm 2

Algorithm 2 Hash-based Method for Apriori

```
Repeat for each transaction of the database
D = set of all possible k-itemsets in the transaction
Description:
for each element of $D$ {
    Find a unique integer $uniq\_int$ using the hash function for $k$- itemset;
    Increment $freq[uniq\_int]$
}
Increment $tran\_pos$
Moves pointer to next transaction until $end\_of\_file$
for($freq\_ind = 0; freq\_ind < length\_of\_the\_array(two\_to\_three\_freq));$
    $freq\_ind++)$ {
    if $(freq[freq\_ind] >= requiredsupport);$
    mark the corresponding $k$-itemset }
```

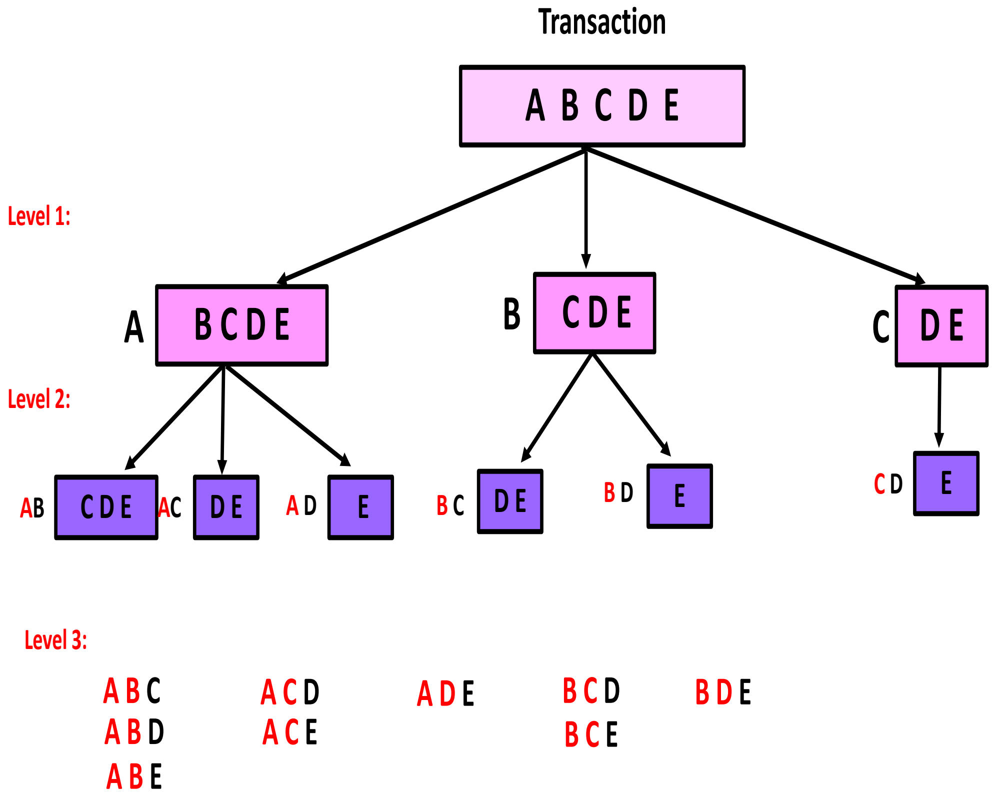


Figure 3.5: Improving Apriori using Hash-based Technique

3.9.1 Improving *Apriori* Using Transaction Reduction Technique

Apriori efficiency can be also be enhanced by eliminating irrelevant transaction records and reducing excessive sub-items generated during pruning of the candidate itemsets [Singh et al., 2013]. The pruned candidate itemsets then form a set of infrequent itemsets therefore removed from the process alongside with any candidate with an infrequent subset.

3.9.2 Partitioning in *Apriori*

Data can be partitioned to detect candidate itemsets. Partitioning method that requires only two scans of the database can be applied to enhance *Apriori* efficiency. As presented in Fig. 3.6, the partitioning comprises of two stages; in the first stage the method splits the transactions in the database *DB* into n widely separated partitions. Any transaction in the *DB* that satisfy the minimum support threshold then count. Partitioning method considers the calculation of its minimum support as follows:

$$min\_sup \times the\ number\ of\ transactions\ in\ that\ respective\ partition$$

With this procedure, every frequent itemsets **(termed local frequent itemsets)** embedded the partition are detected. The local frequent itemset identifier is used as special data construction that records the transaction IDs of the transactions comprising the items in the itemset. This process allows for any local frequent k-itemsets, such that $k = 1, 2, 3, ...$, are detected in a single scan. An item is said to be frequent in the whole DB if it is frequent in at least one partition of the DB. A group of frequent itemsets from all partitions is collected and becomes the **global candidate itemsets** in the entire DB.

In the second stage, another scan is ran to ascertain the real support of each candidate to confirm the global frequent itemsets. Partition size and number are pre-defined to avoid mis-fitting into the main memory.

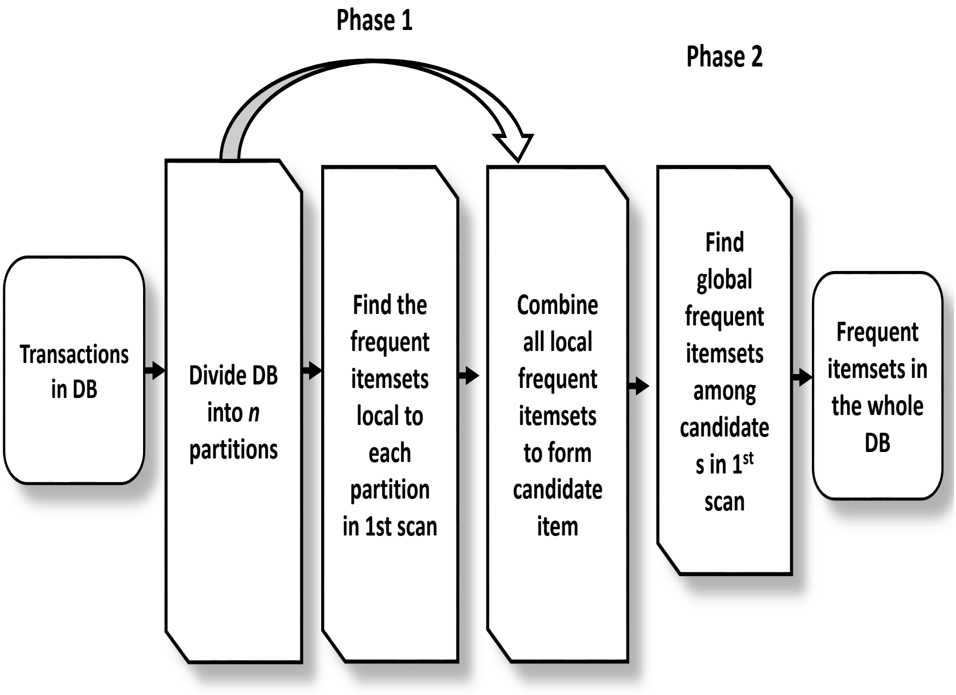
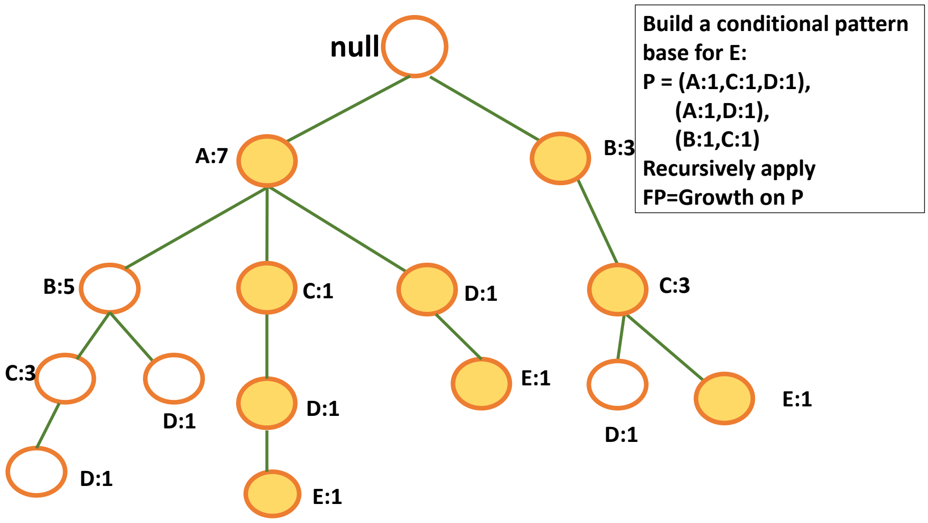

Phase 1
Phase 2
Transactions in DB
Divide DB into *n* partitions
Find the frequent itemsets local to each partition in 1st scan
Combine all local frequent itemsets to form candidate item
Find global frequent itemsets among candidate s in 1st scan
Frequent itemsets in the whole DB

FIGURE 3.6: Improving Apriori using Partitioning Technique

3.10 Frequent Pattern-Tree

Frequent Pattern Tree *(FP-Tree)* algorithm is an essential tool for mining association [Agrawal et al., 1994, Agrawal et al., 1993, Agrawal and Srikant, 1995], relationship [Brin et al., 1997a], occurrences [Mannila et al., 1997], causality [Silverstein et al., 2000], clustering [Lent et al., 1997], chronological patterns [Agrawal and Srikant, 1995], fractional periodicity [Han et al., 1999], emerging patterns [Dong and Li, 1999] and other data mining tasks. *FP Growth* Algorithm is a scalable technique for extracting comprehensive set of increasing frequent patterns using an expanded prefix-tree process for storing compact and vital information relating to frequent pattern tree *FP-Tree* [Han et al., 2000]. It is best to illustrate *FP-Tree* using an example. The technique reads each transaction (as presented in Table 3.8) and then maps them to a path in the

FP-Tree. In a situation where each transaction has an exclusive itemset, an additional space will be required to store the pointers between the nodes and the counter for each item as shown in Fig. 3.7. The complexity of the tree increases as the uniqueness of each transaction increases.



Build a conditional pattern base for E:
P = (A:1,C:1,D:1),
(A:1,D:1),
(B:1,C:1)
Recursively apply FP=Growth on P

Figure 3.7: Diagram Showing *FP-Tree* Process

Table 3.8: Transaction Database for Generating *FP-Tree*

tx	Items
1	f,a,c,d,g,i,m,p
2	a,b,c,f,l,m,o
3	b,f,h,j,o
4	b,c,k,s,p
5	a,f,c,e,l,p,m,n

Let us set the *min_sup* threshold to 3. The algorithm runs through the database at the first go and detects individual items (in itemset) that satisfies the *min_sup*.

TABLE 3.9: Generating Itemsets in *FP-Tree* with *min_supp* of 3

f: 4	~~i: 1~~
a: 3	~~j: 1~~
b: 3	~~k: 1~~
c: 4	~~l: 2~~
~~d: 1~~	m: 3
~~e: 1~~	~~n: 1~~
~~g: 1~~	~~o: 2~~
~~h: 1~~	p: 3

TABLE 3.10: Categorising Frequent Items in Decreasing Order

f:	4
c:	4
a:	3
b	3
m	3
p	3

The database is scanned to ascertain the support count of each item. Infrequent items (items with less than less than *min_sup* of 3) are eliminated and frequent items are categorised in decreasing order as illustrated in Table 3.9. The database is scanned, one transaction at a time to build the *FP-Tree* for each transaction. If it is an exclusive transaction, then a new path is created and the counter for each node is set to 1. If it shares a joint prefix itemset, the joint itemset node counters is increased and new nodes is built if necessary.

This is repeated until each transaction has been mapped unto the tree. Having explained the general concept of *FP-Tree*, next we are going to demonstrate how *FP-Tree* algorithm is constructed based on [Jiawei Han, 2011].

3.10.1 Construction of *FP-Tree* Algorithm

1. Scan the transaction DB once. Assemble F, which forms the set of frequent items, as well as the support of every frequent item. Sort F in support-descending arrangement as *FList*, (list of frequent items.)

2. Build the root of an *FP-Tree*, T , and tag it as “null”;

3. For each transaction *tran* in D do the following:

4. Choose and arrange the frequent items in *tran* in the order of L;

5. Let the arranged frequent item list in *tran* take the form of [p | P];

6. Where p is the first component and P is the subsequent list of items.

7. Call *insert_tree*([p | P], T);
   the function *insert_tree*([p | P], T) is summarised as follows:

8. If T has a child N implying that $N.item\text{-}name = p.item\text{-}name$;

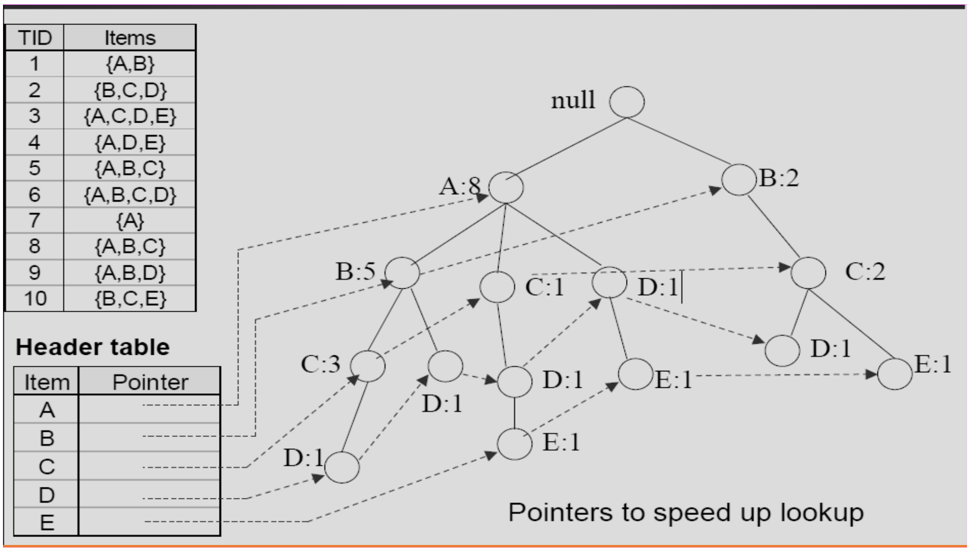
9. Then increment N’s count by 1;

10. Else build a fresh node N, with its count starting from 1, having parent-link connected to T, and its node-link connected to the nodes with the matching *item-name* through the node-link arrangement;

11. If P is non-empty;

12. Then call *insert_tree*(P, N) recurrently.

Fig. 3.8 demonstrates a complete *FP-Tree* for sample transactions.



TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{A}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Header table

Item	Pointer
A	
B	
C	
D	
E	

FIGURE 3.8: Diagram Showing Complete *FP-Tree* for Sample Transactions.

3.11 Disadvantages of *FP-Tree*

Even though *FP-Tree* is known to be a faster method of extracting frequent itemsets in large databases, the algorithm is not suitable for interactive mining structure. It is not flexible when it comes to tuning minimum support threshold to suit the extraction of relevant association rules as required. Changing the minimum support threshold will lead to repeating the entire mining procedure [Kotsiantis and Kanellopoulos, 2006]. This is not the case with *Apriori* as the algorithm allows tuning of minimum support threshold in order to enhance the extraction of strong rules that are relevant to the database under consideration. The minimum support and confidence threshold settings used for the experiments conducted in this thesis were tuned where necessary to adequately model the different datasets employed for the experiments. *FP-Tree* is also disadvantaged in the case of incremental mining. An inclusion of any new dataset may compel a complete replication of the entire process, which is not the case with *Apriori*.

3.12 Why *Apriori* Algorithm?

Apriori is considered a standard algorithm for learning *ARs* [Tjioe and Taniar, 2004]. It extracts *ARs* from transactional databases. The resemblance between transactional databases and tweets and their hashtags has motivated our research as aforementioned in Section 3.4. *Apriori* uses the *breadth-first* search and then a tree construction to count candidate itemsets competently. It creates candidate itemsets of length k from the itemsets of length $k-1$. By pruning the candidates which are infrequent and therefore have infrequent subsets, *Apriori* employs the down closure process to extract all frequent $k-length$ itemsets. *Apriori* algorithm is adopted in the experiments reported in this thesis after considering the following advantages over *FP-Tree*:

- *Apriori* is more efficient during the candidate itemsets generation procedures [Tjioe and Taniar, 2004].

- The algorithm is suitable for running on a parallel processing system [Jin et al., 2005]. Although parallel processing system is not used in this thesis, it allows other users to consider its advantageous aspect when applying *Apriori* to experimental datasets.
- *Apriori* algorithm is easy to implement [Kumar and Rukmani, 2010].
- As mentioned in Section 3.4, *Apriori* algorithm allows for multiple setting of the minimum support threshold without affecting the process of frequent items and rules extraction.
- *Apriori* allows incremental mining. New datasets can be introduced into the mining process at any time.
- *Apriori* capacity to handle the adoption of *time window* introduced in the thesis experimental process enhances the speed of generating the frequent itemsets (hashtag keywords) in each time window. Time window is discussed in details in Chapter 5. *ARs* in each of the time windows are subsequently extracted and mapped to targeted real-life topic/event. *Apriori* capacity of handling non-sparse datasets gives it a leverage over *FP-Tree* whose data structure presents a stumbling block against its parallelization.

3.13 Summary

In this chapter, we presented an overview of *ARM* which, is the technique used to build the methodology used in the experiments conducted in this research. We gave a comprehensive explanation of *ARM* main concepts such as the market basket analysis, support and confidence and the lift. We compared and contrasted between *Apriori* and the *FP-Tree* algorithms. Finally we justified the adoption of the *Apriori* algorithm for our research experiments over the *FP-Tree* algorithm. In Chapter 4 we adopt *Apriori* algorithm to build **TRCM**. We define different types of *TRCM* rules and discuss the main building-blocks of the *TRCM* system that are used for topic detection and tracking *TDT* experiments in this research.

Chapter 4

Transaction-based Rule Change Mining

The findings reported in this chapter have been published in [1]. The report in the paper is presented in this chapter and dully referenced.

4.1 Motivation

There has been numerous experiments conducted on Twitter data to detect relevant topics/events [Becker et al., 2011, Cataldi et al., 2010, Corney et al., 2014, Jackoway et al., 2011, Becker et al., 2012, Kasiviswanathan et al., 2011, Glass and Colbaugh, 2010] as well as topic tracking on the network [Benhardus

[1] Adedoyin-Olowe, M., Gaber, M. M., & Stahl, F. (2013, January). TRCM: A methodology for temporal analysis of evolving concepts in twitter. In Artificial Intelligence and Soft Computing (pp. 135-145). Springer Berlin Heidelberg

and Kalita, 2013, Dong and Li, 1999] in the last decade. However, none of the techniques were used to **detect** and **track** diverse topic/event in Twitter simultaneously. Since most real-life topics/events evolve over a period of time, and sometimes result in the occurrence of other topics/events, it is necessary to propose a technique that is capable of not only detecting topics/events on Twitter but to also track the evolvement of such topic/events over specified period of time. This will remove the complexity of developing other technique(s) to track already detected topic/event on Twitter network. Other techniques mine entire corpus of Twitter posts for Topic Detection and Tracking *TDT*, while very little attention is given to the analysis of tweet hashtags which are known to give title to tweets and describe their contents while also enhancing their readability. Our research method detects and also tracks the evolvement of targeted topic/event over a specified period by applying Association Rule Mining *(ARM)* of the hashtag keywords present in evolving tweets. Our method is able to address the complexity of separating topic/event detection and tracking in Twitter.

In this chapter, we introduce the research methodology used for the experiments conducted in this thesis and explain how it analyses tweets on the same topic over consecutive periods t and $t+1$.

4.2 Rule Dynamics of Association Rule Mining

As explained in Section 3.4 of Chapter 3, an association rule is in the form $X \Rightarrow Y$, where X and Y are disjoint sets of items. Hashtag keywords are viewed as items and the tweets containing the hashtags are viewed as transactions in the database (as presented in Table 3.4) of Chapter 3. The capability of *ARM* technique enables it to uncover different patterns in both transactional and relational datasets. Changes in rules dynamics patterns generated using the ***Apriori*** algorithm of *ARM* can be used for *TDT* on Twitter network. An example of this is breaking news of a disaster, say an earthquake in Japan. The news will tend to generate strong rules in tweets at the early stage. This is referred to as speedy rule emergence. The emergence of this rule can result in the broadcast of the incident as breaking news by news agencies in real-life. It can also help other organisations like the **"Red Cross"** to respond swiftly and dispatch aids to the affected areas. **Transaction-based Rule Change Mining *(TRCM)*** is the methodology proposed in this research. It applies *Apriori* to hashtags present in tweets at subsequent time periods *t* and *t + 1* as presented in Fig. 5.3 and produce two association rulesets which are interpreted as rule evolvement in the context of this research. Rules evolvement is explained in details in Sections 4.6 and 4.7.

4.3 Transaction-based Rule Change Mining Architecture

TRCM is a system built to identify rule change patterns in tweets at different period of time. The application of *Apriori* method of *ARM* to hashtags in tweets at *t* and *t + 1* generates two association rulesets. In [Adedoyin-Olowe et al., 2013] *TRCM* was used to detect four (temporal) dynamic rules in tweets. The four rules identified are namely "**new**" rules, "**unexpected**" rules, "**emerging**" rules and "**dead**" rules. The rules were obtained by matching rules present in tweets at t and $t+1$. The **Rule Matching Threshold *(RMT)*** were represented with binary vectors $[0,1]$, with 0 indicating the non-existence hashtag(s) in tweets, while 1 indicates the existence of hashtag(s) in tweets. Degree of similarity and difference measures are applied to detect rule change in tweets as presented in Fig. 4.1. The changes are categorised accordingly under the four identified rules. *TRCM* reveals the dynamics of *ARs* present in

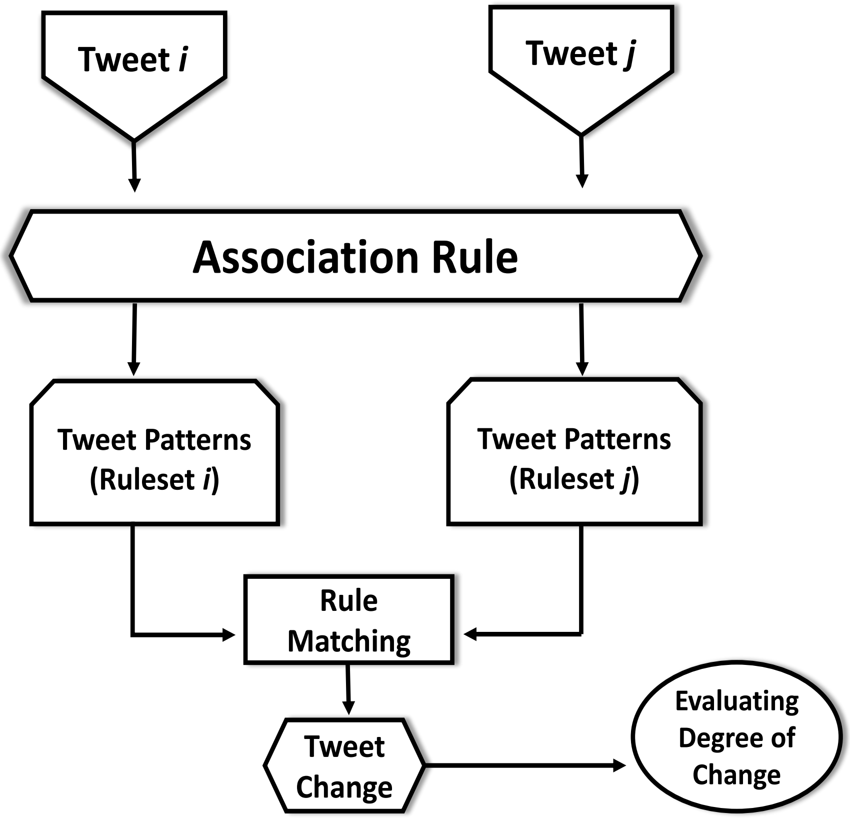

Tweet *i*
Tweet *j*
Association Rule
Tweet Patterns (Ruleset *i*)
Tweet Patterns (Ruleset *j*)
Rule Matching
Tweet Change
Evaluating Degree of Change

FIGURE 4.1: The Process of Tweet Change Discovery.

tweets and demonstrates the linkage between the different types of rule dynamics investigated. The rules at t and $t + 1$ are matched using **Rule Matching (RM)**. *RM* is the process of matching the *right hand side/consequent* and the *left hand side conditional* part of the *ARs* in itemsets at time t and $t + 1$ to detect hashtags at $t+1$ that has any similarity with those at t having considered the user-defined *RMT*. The adoption of *RM* to the two itemsets result in the detection of the four identified rules patterns present in tweets' hashtags.

4.3.1 Rule Similarities and Differences

Song et al [Song et al., 2001] and Liu et al [Liu et al., 2009] used similar methods for calculating similarities and differences between two rules in relational datasets to detect association rules at two-time periods. Song et al [Song et al., 2001] developed **similarity** and **difference** measures for rule matching to automatically detect changes in customer behaviour using customer profiles and sales data at different periods. Liu et al [Liu et al., 2009] improved on the adaptation of the similarity and difference measures proposed by Song et al [Song et al., 2001] to mine the change of event trends for decision support in environmental scanning. While the former focuses on unexpected consequent changes without considering the unexpected condition changes, Liu et al [Liu et al., 2009] consider the attributes in both the consequent and the conditional parts of the rules and further measured the degree of the unexpected changes. Our research method improved on the work carried out in both Song et al [Song et al., 2001] and Liu et al [Liu et al., 2009] by:

- applying *ARM* on tweets' hashtags at 2 consecutive periods to detect real-life topics/events from diverse domains such as politics, sports, social-economic and business; [Adedoyin-Olowe et al., 2013, Gomes et al., 2013, Adedoyin-Olowe et al., 2014b, Adedoyin-Olowe et al., 2014a, Adedoyin-Olowe et al., 2015];

- tracking rule evolvements on Twitter using *Rule Trend* approach and mapping the rule evolvements to real-life topics/events/news [Adedoyin-Olowe et al., 2015]; and

- adopting the mathematical model by Song et al [Song et al., 2001] and Liu et al [Liu et al., 2009] to transaction databases represented in our research in form of tweets [Adedoyin-Olowe et al., 2013, Gomes et al., 2013, Adedoyin-Olowe et al., 2015].

Details and explanations of the calculations and notation used for the development of *TRCM* are stated in Section 4.3.2.

TABLE 4.1: Notation of Terms

n	number of hashtags
r_i^t	a set of all rules generated at time t where $i \in \{1,..., \mid r^t \mid\}$
r_j^{t+1}	a set of all rules generated at time $t+1$ where $j \in \{1,..., \mid r^{t+1} \mid\}$
lh_i/lh_j	number of hashtags with value 1 present in conditional part of rule i and j
rh_i/rh_j	number of hashtags with value 1 present in consequent part of rule i and j
lh_{ij}/rh_{ij}	number of matching hashtags in conditional/consequent part of rules i and j
p_{ij}/q_{ij}	degree of similarity of hashtags in conditional/consequent part of rules i and j
thp_{ij}/thq_{ij}	Threshold of degree of similarity of hashtags in conditional/consequent part of rules at t and $t+1$

The notation used for developing the mathematical model of *TRCM* is given in Table. 4.1.

4.3.2 Measuring Similarity

$$p_{ij} = \frac{lh_{ij}}{max(lh_i, lh_j)}(1)$$

$$q_{ij} = \frac{rh_{ij}}{max(rh_i, rh_j)}(2)$$

TRCM framework defines rule change patterns in tweets at different periods of time. Rules in new ruleset (r_j^{t+1}) are matched with those in old ruleset (r_i^t) to detect the emerging rules and unexpected rules. On the other hand, rules in old ruleset (r_i^t) are matched with those in new ruleset r_j^{t+1} to confirm rules that are dead and therefore are no longer present in r_j^{t+1}. The steps of measuring **Tweet Change Discovery (TCD)**, are explained next with real-life examples.

Step 1: For each rule in the conditional part of the new ruleset r_j^{t+1} as shown in Fig 4.3, match with each rule in the conditional part of the old ruleset r_i^t to identify similarity. Where similar hashtag is identified, compute the number of hashtags that appear in the conditional parts of both rulesets. In Fig 4.2 we identify #KDD, #excel from r_j^{t+1} matching with the rules in the *lhs* of r_i^t, while *#datamining* and *#KDD* are found to be similar in the *rhs* of r_i^t.

Step 2: Divide the number of similar hashtags in the *lhs* of r_j^t; (#KDD, #excel) by the maximum number of hashtags in the *lhs* of both rules that were currently being matched in the conditional parts of either the old or new ruleset. For example in the conditional part of ruleset r_j^{t+1} (new ruleset), 2 hashtags were matched and the maximum number of hashtags in both rules are 8 (max of 6 and 8) . In the *lhs* the 8 hashtags are; #KNN, #sqlserver, #excel, #KDD, #bigdata, #analytic, #Facebook and #Privacy, which were found to be similar to those in r_i^t (old ruleset).

$$p_{ij} = \frac{lh_{ij}}{max(lh_i, lh_j)} = \frac{2}{6,8} = 0.25 \tag{4.1}$$

Step 3: Apply the same method in step 2 to the consequent parts of the two rulesets to detect the q_{ij}. In the consequent part of ruleset r_j^{t+1} (new ruleset), 2

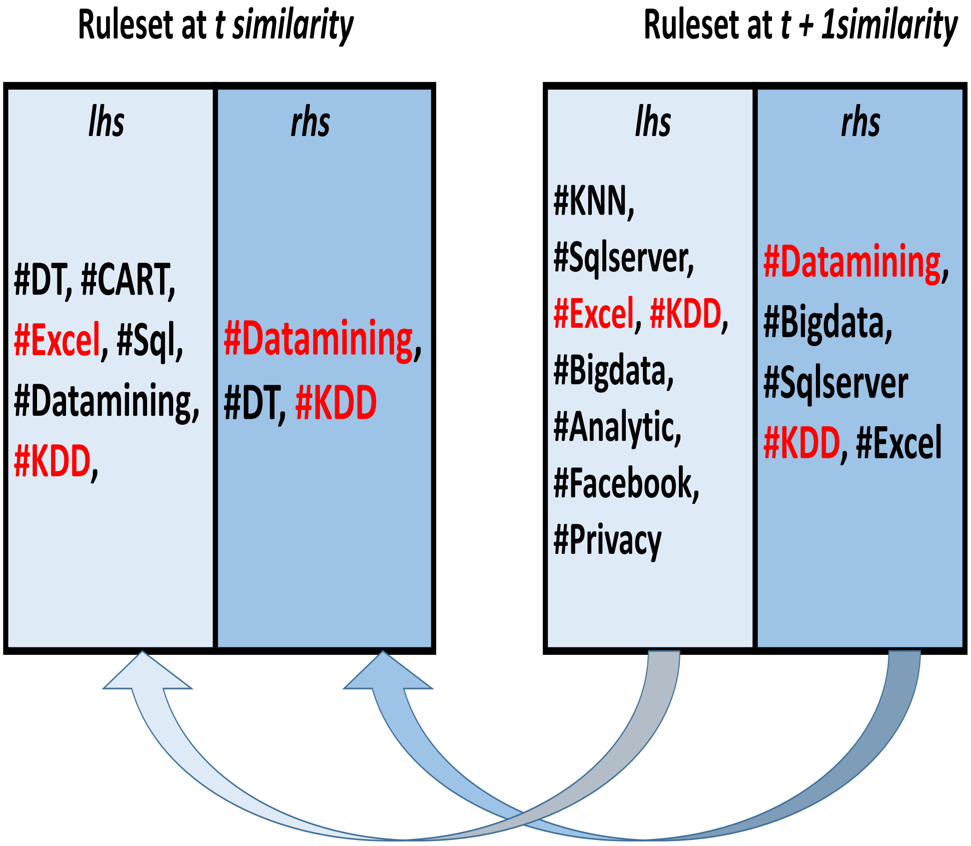
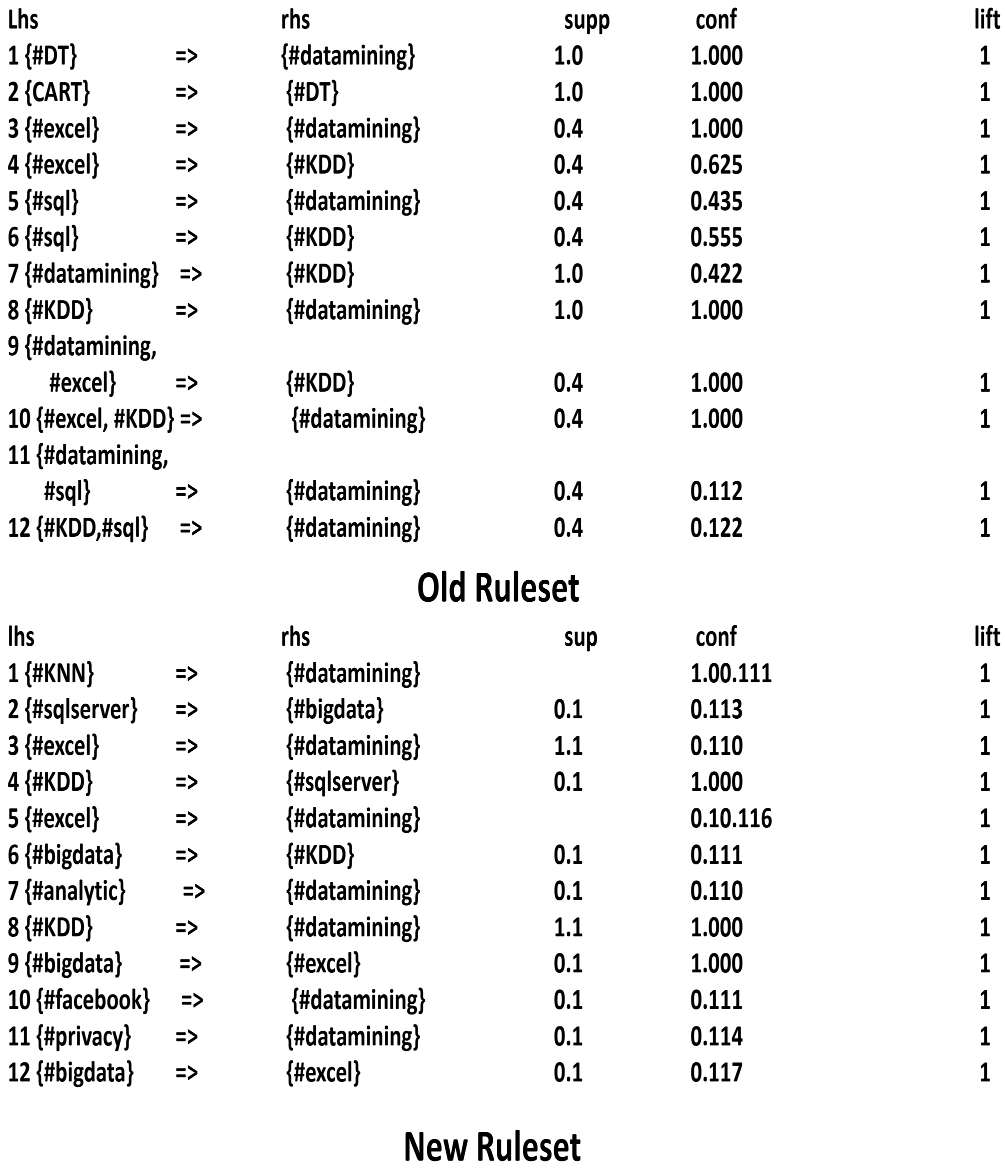


FIGURE 4.2: Rules Matching Illustration

(#DT and #CART) out of the 5 hashtags were matched as similar with those in r_i^t (old ruleset) . Divide the similar hashtags with the maximum number of hashtags as in step 2. The calculations are presented in equation 4.1 and 4.2.

$$q_{ij} = \frac{rh_{ij}}{max(rh_i, rh_j)} = \frac{2}{2,5} = 0.4 \tag{4.2}$$

Step 4: Identify the degree of similarity of rules in the old and new rules. However, for two rules to be similar, their degree of similarity must be greater than the $RMT \in [0, 1]$. In this case, let us define the RMT as 0.5. Where the degree



Lhs		rhs	supp	conf	lift
1 {#DT}	=>	{#datamining}	1.0	1.000	1
2 {CART}	=>	{#DT}	1.0	1.000	1
3 {#excel}	=>	{#datamining}	0.4	1.000	1
4 {#excel}	=>	{#KDD}	0.4	0.625	1
5 {#sql}	=>	{#datamining}	0.4	0.435	1
6 {#sql}	=>	{#KDD}	0.4	0.555	1
7 {#datamining}	=>	{#KDD}	1.0	0.422	1
8 {#KDD}	=>	{#datamining}	1.0	1.000	1
9 {#datamining, #excel}	=>	{#KDD}	0.4	1.000	1
10 {#excel, #KDD}	=>	{#datamining}	0.4	1.000	1
11 {#datamining, #sql}	=>	{#datamining}	0.4	0.112	1
12 {#KDD,#sql}	=>	{#datamining}	0.4	0.122	1

Old Ruleset

lhs		rhs	sup	conf	lift
1 {#KNN}	=>	{#datamining}		1.00.111	1
2 {#sqlserver}	=>	{#bigdata}	0.1	0.113	1
3 {#excel}	=>	{#datamining}	1.1	0.110	1
4 {#KDD}	=>	{#sqlserver}	0.1	1.000	1
5 {#excel}	=>	{#datamining}		0.10.116	1
6 {#bigdata}	=>	{#KDD}	0.1	0.111	1
7 {#analytic}	=>	{#datamining}	0.1	0.110	1
8 {#KDD}	=>	{#datamining}	1.1	1.000	1
9 {#bigdata}	=>	{#excel}	0.1	1.000	1
10 {#facebook}	=>	{#datamining}	0.1	0.111	1
11 {#privacy}	=>	{#datamining}	0.1	0.114	1
12 {#bigdata}	=>	{#excel}	0.1	0.117	1

New Ruleset

Figure 4.3: Rules Matching Sample

of similarity is less than 0.5, the rules are considered to be different (new rule or dead rule as shown in Fig. 4.4). With the foregoing, the computation of the p_{ij} and q_{ij} in our sample datasets shows similarity degree in both the conditional and the consequent parts of the rules with the 0.25 and 0.4 less than *RMT*. The left hand side $(lhs)/conditional$ and the right hand side $(rhs)/consequent$ parts of rules in *Apriori* method is used to analyse hashtags as conveyed in tweets over a defined period of time. The co-occurrence of frequent hashtags is used to detect Association Rules *(ARs)* present in tweets at different periods of time. The similarities and differences in the *ARs* discovered in tweets at time t and $t+1$ are measured in order to categorise them under a rule pattern (for example emerging rule). Emerging rule detection such as breaking news of a disaster like Typhoon in the Philippines can trigger an instantaneous action from disaster emergency response organisations as described in Section 4.2.

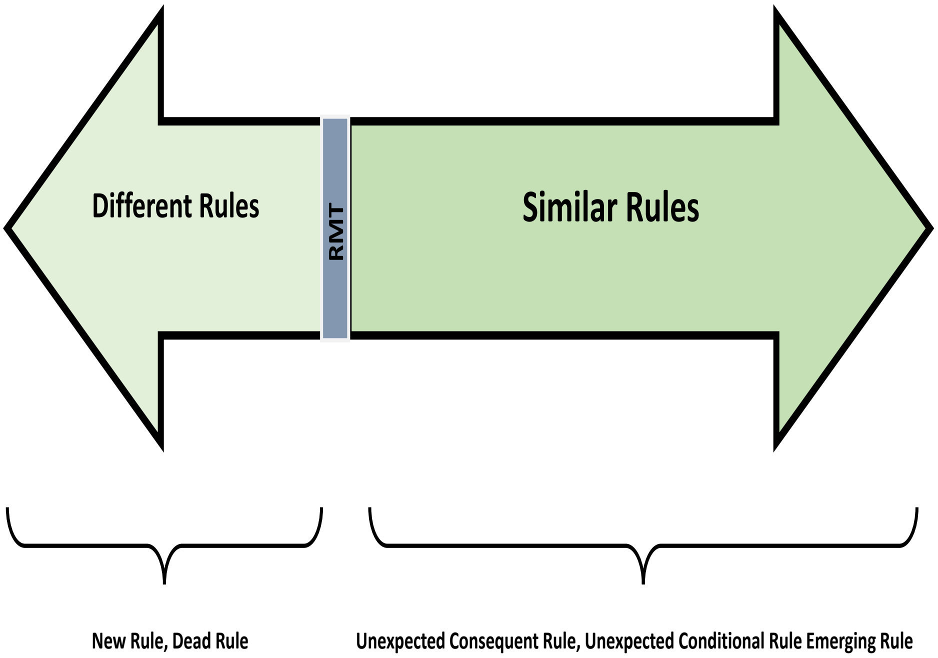

Different Rules
RMT
Similar Rules
New Rule, Dead Rule
Unexpected Consequent Rule, Unexpected Conditional Rule Emerging Rule

FIGURE 4.4: Rules Similarities and Differences

4.4 Definitions of *TRCM* Rules

Rule Matching in rulesets at t and $t+1$ results in the definition of *TRCM* rule change patterns. **Unexpected Consequent rule** arises when a rule in r_i^t and another rule in r_j^{t+1} have similar conditional part but different consequent part ($p_{ij} \geq thp_{ij}$ and $q_{ij} < thq_{ij}$) as presented in Fig. 4.5.

- $\#flightMH370 \Rightarrow \#missing$ **(Rule at time** t**)**
- $\#flightMH370 \Rightarrow \#TimAkers$ **(Rule at time** $t+1$**)**

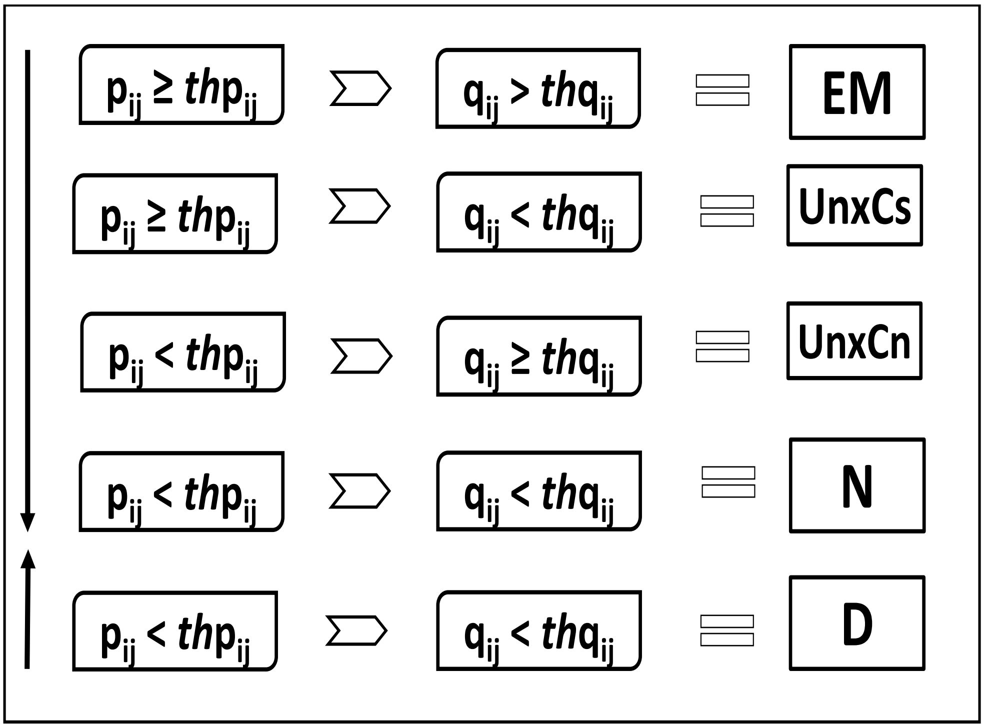


FIGURE 4.5: TRCM Rules Assignment

Unexpected Conditional rule is detected when the consequent parts of rule r_i^t at and r_j^{t+1} are similar, but the conditional parts are different ($p_{ij} < thp_{ij}$

and $q_{ij} \geq thq_{ij}$). The similarity measure must be greater than or equal to the user-defined *RMT*. Having described unexpected consequent rule change in real-life situation, it is important to mention that both unexpected consequent and unexpected conditional rule change are presented in the same way in real-life. An example of unexpected rule in real-life is sudden event occurrence, in the case of the missing Malaysia flight, claim by Tim Akers, a British marine archaeologist of having found flight MH370 3,000 miles from the search zone after spotting debris painted in the colours of Malaysia Airlines can result in unexpected rule change.

- $\#Malaysia \Rightarrow \#flightMH370$ **(Rule at time** t**)**
- $\#Missing \Rightarrow \#flightMH370$ **(Rule at time** $t+1$**)**

Emerging rules occur when rules at time t and $t+1$ have similar conditional and consequent parts of the rule with similarity greater than the user-defined threshold ($p_{ij} \geq thp_{ij}$ and $q_{ij} > thq_{ij}$). An instance of a real-life event that may generate an emerging rule in *TRCM* is global breaking news of a disaster or the announcement of the US presidential elections winner.

- $\#Missing \Rightarrow \#flightMH370$ **(Rule at time** t**)**
- $\#Missing \Rightarrow \#flightMH370$ **(Rule at time** $t+1$**)**

Breaking news often evolve into emerging rules within a short period of time due to the volume of tweets hashtagging major keyword relating to the news. All rules at $t+1$ that were not classified as one of the three previous types of rules (emerging, unexpected consequent and unexpected conditional rules) are classified as **new** rules. This means that all rules in ruleset at $t+1$ are new until there is a match found in ruleset at t. A rule in t is classified **dead** if its maximum similarity measure with all the rules in $t+1$ is less than the user-defined *RMT* from both the conditional and consequent parts. “Dead” rules in real-life are topics that were initially tweeted but are no longer visible in Twitter network after some time. The display of an initial status is referred to as a *reverse trend*. However, while most rules end up being dead, some may not, which means that such rules are still active on Twitter even though they may cease to evolve (static rule).

4.5 TRCM-Rule Type Identification

TRCM-Rule Type Identification *(TRCM-RTI)* is a technique based on TRCM. It is applied to discover *rule trend* of tweets’ hashtags over a consecutive periods. Rule Trend demonstrates how rules patterns evolve into different Time Frame Windows *(TFWs)* and the length at which they remain in the same status in each time frame window as shown in Fig 4.7. Simply put, *TFW* is used to measure the lifespan of specific hashtags on Twitter in relation to the evolvement of related topic/event in reality.

TFW reveals rule evolvements in tweets at different periods during the lifespan on Twitter network. It also calculates the length at which rules maintain a status on the network. This process is referred to as **rule trend** within different *TFWs*. *TFW* is explained in details in Sections 4.6 and 4.7.

4.6 Rule Trend Analysis in Tweets

Trend Analysis *(TA)* of tweets is a way of inspecting the progression of rules present in tweet hashtags over a temporal period of time. The ultimate goal of *TA* is to be able to trace back the origin of a rule (rule trace). A rule $X \Rightarrow Y$ may have actually started up as $A \Rightarrow B$ and over time the rule has evolved unexpectedly as presented in Fig. 4.6. The time frame between $X \Rightarrow Y$ and $A \Rightarrow B$

may vary depending on factors that may affect the rule status at different point of evolvement. *TFW* describes the different evolvement chains/sequences rule status any tweets hashtags is measured in throughout its lifespan on the network. Factors affecting the size of the time frame of rules include unexpected discovery relating to an on-going event. Such discovery may elongate or truncate the lifespan of a hashtag. In [Gomes et al., 2013] *TRCM-RTI* was applied to learn the rule trend of tweets' hashtags over a sequential period. *TFWs* are created to show the different rule evolvement patterns which can be applied to evolvements of news and events in reality. *TFWs* are employed to calculate the lifespan of specific hashtags on Twitter and to link the hashtags to lifespan of related topics/events in real-life. Using the experimental study results, it was established that the lifespan of tweets' hashtags could be linked to evolvements of related topic/event in reality. Rule trend analysis can be demonstrated using the chain/sequence process. This process shows that rules evolve differently depending on the evolvements of related real-life topics/events. A rule may evolve by taking on a different status in every *TFW*, while another rule may retain a status for more than one *TFW* in a consecutive evolving period. In some other trend, a rule may evolve back to assume its former status during the course of evolvements. While most rules end up being dead, some may not; such rules will still be present on Twitter network even though they may become static. Evolving rules are synonymous to updates on trending topics and the pattern of evolving rules could be linked to evolvement of news updates of a breaking news as the event unfolds. Different evolvements of rule patterns of *TA* in tweets are demonstrated in the formalisation in Table 4.2.

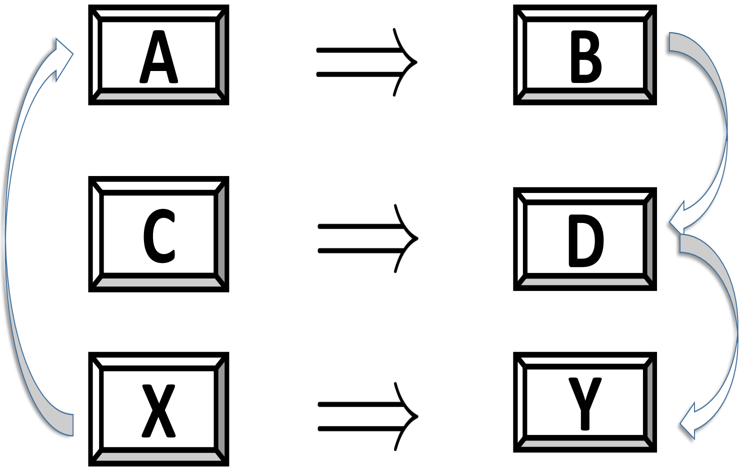


Figure 4.6: Diagram showing Rule Trend

4.7 Time Frame Window of Evolving Rules in Twitter

Time frame plays an important role in Trend Analysis of tweets. While a rule may take a short period to evolve from a new rule to an unexpected rule, another rule may take a longer time to evolve from one rule pattern to another. On the other hand, a rule may become 'dead' straight after becoming a new rule. Such a rule would present a single TFW window (new - 'dead'). As TFW is important to trend analysis of tweets, so it is to news updates in real-life situations. In Fig. 4.7 Sequence A shows an example of how a rule evolved over a time frame period of 47 days and in 4 TFWs before it became a 'dead' rule. The rule has a time frame sequence of C_tN, $C_tU_t^i$, C_tE, C_t^jt, C_tE, C_tD. However, the rule evolved back into C_tE before it became 'dead' after evolving over 5 $TFWs$. In sequence B, the rule started as new rule for 3 days and then evolved into

Table 4.2: Evolving Rules Patterns

T	The total time period intervals a rule status is measured in.
C_t	The category of the rule
C_tN	New rule
$C_tU_t^i$	Unexpected conditional rule
C_t^jt	Unexpected consequent rule
C_tE	Emerging rule
C_tD	Dead rule
TFW	Number of frame window

$C_tU_t^i$. It retained the status for 30 days before it disappeared from Twitter network after evolving over 2 *TFWs*. Lastly, sequence C shows that the rule's first status on Twitter network was retained throughout the specific period of the trend analysis. It did not go into the 'dead' rule state, but remained static on the network. All the sequences in Fig. 4.7 explain how topics/events in reality affect the dynamics of rules. It also shows the importance of some events in real-life when their sequence of evolvements is considered and how long they retain some status. *TFWs* is applied to *TA* of *ARs* identified in hashtag keywords of the experimental datasets analysed in Chapter 5 of this thesis. Understanding the *TA* of rule evolvements in tweets hashtags enables different entities to understand tweets better and make advantageous use of its contents, either as decision support tool or for information retrieval.

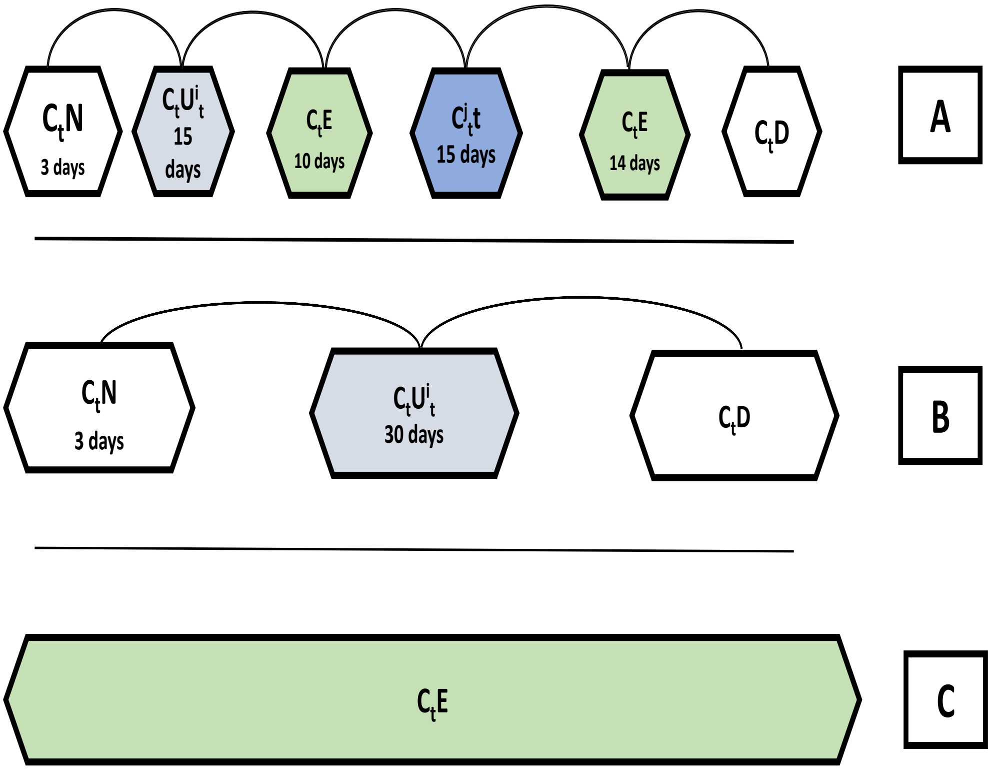


FIGURE 4.7: Time Frame Sequences of Evolving Rules

4.8 Summary

In this Chapter we introduced our research methodology termed *TRCM* architecture. We gave an overview of rule dynamics of *ARM* and explained how we adopt the technique to extract *ARs* present in tweets' hashtags. The theory behind rule similarities and differences was discussed and the calculation of similarity measure was presented. *TRCM* rules was defined and real-life examples were given. We explained how the different *TRCM* rule types are identified. We discussed how *TRCM rules* in tweets' hashtags undergo different trend within specific *Time Frame Windows* and explained the relevance of *Rule Trend Analysis* in tweets. Finally, we gave real-life examples of tweets evolvements on Twitter network. Given that we have explained the theoretical under-pin of our research detailing the components of *TRCM* system with mathematical formalisation, Chapters 5 and Chapters 6 are devoted to both **quantitative** and **qualitative** experiments to validate the proposed methods.

Chapter 5

An Experimental Study of ***TRCM*** for Event Detection

The findings reported in this chapter have been published in [1]. The outcomes of the experiments conducted in the paper is presented in this chapter and dully referenced.

5.1 Introduction

In chapter 4 we discussed our proposed method for event detection and tracking on Twitter. We presented an overview of Association Rule Mining *(ARM)* and explained the adoption of the technique for the extraction of association rules

[1] Adedoyin-Olowe, M., Gaber, M. M., Dancausa, C. M., & Stahl, F. (2014, December). Extraction of Unexpected Rules from Twitter Hashtags and its Application to Sport Events. In Machine Learning and Applications (ICMLA), 2014 13th International Conference on (pp. 207-212). IEEE

(ARs) embedded in tweets' hashtags. We discussed rule similarities and differences and explained their application in this research. We also indicated how similarity measures are calculated. We defined *TRCM* rules and demonstrated how they can be applied in real-life situation. We specified how *TRCM* rules evolve over different time frame windows *(*TFWs) and how rule trend analysis is applied to rule evolvements.

In this chapter we automate the detection of real-life topics generated in 3 tweets datasets from 2 diverse domains; sports (the English FA Cup Final 2012) and politics (US Presidential Elections 2012 and US Super Tuesday 2012). We map all hashtag keywords extracted by our system during training process to related topics from the ground truth to ascertain a match and subsequently to validate our system performance. A match is said to have occurred if the time-slot of extracted hashtag keyword in the specific tweet correlates with the time of event occurrence in the ground truth. We evaluate how the dynamics of each dataset affects our experimental results. Sports events (especially football) is a short-term and relatively emergent event, while political events are long-term and stable event. For performance effectiveness analysis of our method, we apply precision and recall, for this application domain, precision is a more important matrix than recall. This is because we are more concerned with generating relevant hashtag keywords (precision) that are related to targeted real-life topics/events. As far as we are aware of, *TRCM* is the only method that detects all the types of real-life topics/events on Twitter at **once** with the same method using hashtags and *ARM* as previously discussed in Chapter 2.

5.2 Data Extraction and Filtering

The experiments conducted in this chapter extract real-life (newsworthy) topics from tweets' hashtags in the sports and political domain. Hashtags are principally meant to emphasise significant keywords in tweets or give title to posted on Twitter. Since the inclusion of hashtag to keywords in tweets enhance the chances of the readability of such tweets, it is necessary to develop a system that: 1) will extract newsworthy topics from hashtag keywords included in tweets and map them to related real-life topics; and 2) track the evolvement of the topic overtime. The combined use of **hashtags** and **ARM** is a novel TDT method when compared to existing TDT methods used for analysing Twitter data.

To detect newsworthy topics from tweets, datasets collected by [Aiello et al., 2013] was used for the experiments. They began crawling of the Twitter API by supplying hashtag keyword(s) that best describe the targeted topic/event to Twitter streaming API to collect tweets with their metadata, however, for this experiment, we filter only hashtag tweets.

It has become common in recent times for different entities, including events organisers and newsagents, to provide an official hashtag that describes tweets related to their event, for example $\#Supertuesday$, $\#Elections2012$ and $\#FAcup$ were official hashtags of the datasets used in our experiments. Other major hashtag keywords were also supplied to the Twitter API alongside the official hashtags to extract relevant tweets. In the case of the *English FA Cup*, Aiello et al also supply names of major team players of each of the two teams to extract relevant tweets from Twitter.

5.2.1 Association Rule Mining of Tweets' Hashtags

For this experiment, we choose a low minimum support (min_sup) and minimum confidence (min_conf) (0.001). This is for the purpose of eliminating the issue with higher value of support and confidence threshold which overlooks *not-so-frequent* but relevant items in the datasets. The application of *Apriori* to hashtags present in tweets at two time periods t and *t + 1* as presented in Fig. 5.1 produces two association rulesets which we interpret as rule evolvements in the context of this work. In [Adedoyin-Olowe et al., 2013], *TRCM* was used to identify four (temporal) dynamic rules in tweet hashtags (as presented in 4.4 of Chapter 4) namely; "New rules" (N), "Unexpected Consequent" rules (UnxCs)/ "Unexpected Conditional" Rules (UnxCn), "Emerging" rules (EM) and "Dead" rules (D). The rules were obtained by matching rules present in tweets at the two time periods (t and $t+1$). Rule Matching Threshold *(RMT)* for degree of similarity in the conditional part of rules (p_{ij}) and in the consequent part of rules (q_{ij}) are assigned between 0 and 1, with 1 indicating maximum rule similarity and 0 indicating maximum rule dissimilarity as presented in equation. 5.1.

$$RMT = thp_{ij}, thq_{ij} \tag{5.1}$$

$$p_{ij} \in [0,1], q_{ij} \in [0,1]$$

$$Where : i \in \{1,2,3,\ldots|r^t|\}$$

$$j \in \{1,2,3,\ldots|r^{t+1}|\}$$

The degree of similarity/dissimilarity measures is built to detect change in rules. The changes are then grouped under the four identified *TRCM* rules. *TRCM* reveals the dynamics of *AR* present in tweets and demonstrates the linkage between the different types of **rule evolvements/trends** as discussed

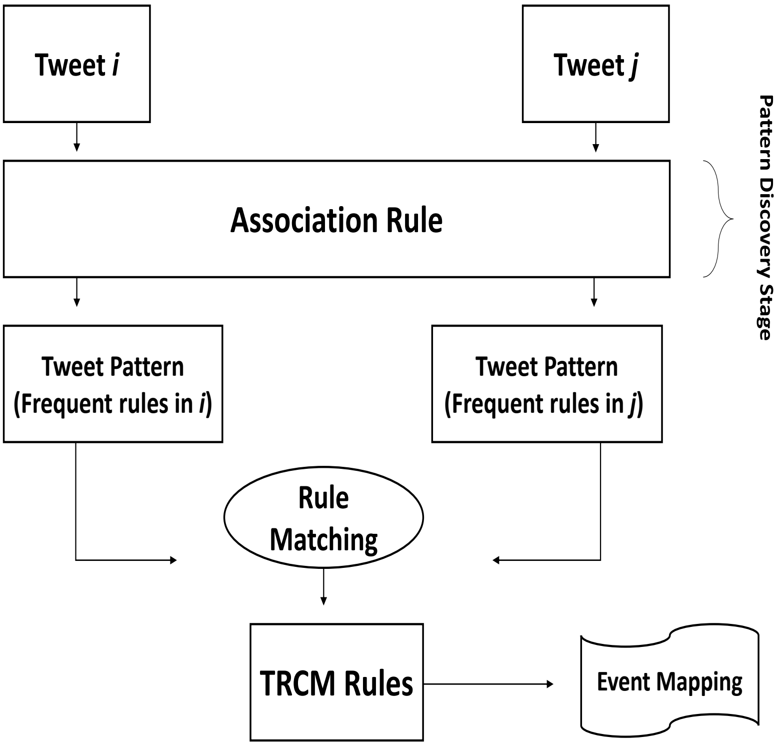


FIGURE 5.1: Event Mapping Process

in Section 4.6 of the previous chapter. It is noteworthy to mention that, for two rules to be similar, their degree of similarity must be greater than or equal to the pre-defined Rule Matching Threshold (RMT).

$$\text{Similarity Measure} = \text{Degree of similarity between } r_i^t \text{ and } r_j^{t+1};$$

$$(0 \leq p_{ij} \leq 1,\ 0 \leq q_{ij} \leq 1) \tag{5.2}$$

Where degree of similarity is less than the RMT, the rules are said to be different.

In this chapter, we extract different *TRCM* rules present in each of the three datasets used for our experiments. First, we analyse the datasets by extracting only hashtag keywords that falls under the unexpected rules (consequent and conditional). Next, we extract only hashtag keywords that falls under the emerging rules. Finally, we analyse the datasets by extracting hashtag keywords that fall under the unexpected and emerging rules combined. The results evaluate the performance profile of each of the *TRCM* rules when applied autonomously on the datasets and the degree of performance enhancement when both sets of rules are combined namely: *Unexpected* and *Emerging* rules.

5.3 Trend Analysis of Identified Rules

Experimental investigations conducted in [Adedoyin-Olowe et al., 2013, Gomes et al., 2013] show that *ARs* present in tweets, hashtags evolve over time. This resulted in what is referred to as **rule trend**. Trend Analysis *(TA)* in the context of the research reported in this thesis, is a way of analysing the trend (evolvements) of *TRCM* rules identified in tweets as displayed by hashtag keywords over a specified period of time. The process of *TA* provides the ability to trace back the root of *TRCM* rules as they evolve on Twitter. This process is called *rule trace*. In the case of the *US Presidential election* dataset, an unexpected rule $\#HealthCare \Rightarrow \#HealthInsurance$ may be traced back to $\#US \Rightarrow \#Obama$. The time frame between $\#HealthCare \Rightarrow \#HealthInsurance$ and $\#US \Rightarrow \#Obama$ may vary depending on different factors that might have affected the rule's status at different point in time. Time Frame Windows *(TFWs)* for the US Presidential election describe the different rule evolvements stages the respective hashtag keywords evolve over during their lifespan on Twitter. Their evolvements are characterised by different occurrences such as the passing of the health care bill which has been a subject of debate in the

US for many year. The bill was finally passed into law in 2010. The US Presidential elections of 2012 however led to the frequent use of $\#HealthCare$ and $\#HealthInsurance$ on Twitter which suggest the reiteration of the health care issues at the time.

5.4 Rule Dynamics in Different Domain and Time Frame Setting

Some rules evolve rapidly (within minutes) while others take longer period to evolve (days or months) depending on the domain which it belongs. A rule #*Drogba* =>#goal in a football event may evolve into #*Drogba* =>#yellowcard within the next minute. This evolvement implies that Drogba scored a goal and in the next minute, he was booked for foul play. In politics, a rule #*Obama* ⇒ #*Ohio* may take 5 hours to evolve into #*Obama* =>#victoryspeech. In this case, the first rule may have been detected when Obama won the poll in Ohio and the second rule detected when he gave his victory speech five hours later. A rule may start up as an emerging rule based on the dynamics of the topic involved (for example breaking news), another rule may display only the "New" status and become "Dead" shortly afterwards.
To this end, TFW setting should be smaller for high dynamic events, on the other hand, it should be set at a higher value for less dynamic events.

5.5 Aims of the Experiment

The main focus of our experiments is to extract real-life (newsworthy) topics from tweet hashtags of any domain using *TRCM*. This requires the development of a framework that will serve as a TDT tool for extracting evolving ARs of hashtags from event tweets and mapping them to the ground truth within relevant time window. The combined use of *hashtags* and *ARM* is a novel *TDT* method when compared to existing *TDT* methods used for analysing Twitter data.

5.6 Methodology

Our methodology process begins from the description of datasets used for the experiment as described in 5.6.1.

5.6.1 Datasets

5.6.1.1 The English FA Cup Finials 2012

The English Football League is a popular and important tournament in English football games. The event is viewed all over the globe with fans of English football clubs spanning all around the world. Participation in the tournament is available to all teams who took part in the Premier League, the football League as well as the all the 5 stages of the FA National League System. Some nominated teams in the stage 6 are also allowed to take part. The tournament marks the peak of the several divisional leagues with the winners of each division advancing to participate in the FA Cup finals. This tournament is known to be the oldest association football competition in the world which dates back to 1871 [Aiello et al., 2013]. Tweets collected for the experiments considered the official hashtag #$FACup$2012 and other main hashtag keywords such as club names of the two clubs #*chelsea* (and #*CFC*), #*Liverpool* (and #*LFC*) and names of big player from the two teams such as #*Drogba* and #*Gerrard*. The 2012 FA Cup finals featured Chelsea Football Club and Liverpool Football

Club, with both teams having huge amount of fans in and outside the UK. Considering the fact that the date chosen for the tournament in the previous year clashed with the champions League Final, the 2012 fixtures are organised to take place within four-week after the end of the English season and the beginning of the UEFA Euro 2012. Expectedly, fans of the two teams tweeted about the match before, during and after the match was played. Chelsea Football Club won by 2 goals to 1.

5.6.1.2 The US Presidential Election 2012

The US Presidential Election 2012 was the 57^{th} four-yearly presidential election. It was conducted in November 2012 with Barak Obama (the incumbent president and his running mate Vice President Joe Biden) representing the Democratic Party and the former Governor of Massachusetts Mitt Romney (and his running mate Representative Paul Ryan of Wisconsin) representing the Republican Party. The election results reinstated the incumbent US president and his running mate for a second term in office. Tweets collected for the experiments considered the official hashtag $\#Election2012$ and the names of the two presidential candidates $\#Obama$ and $\#Romney$.

5.6.1.3 Super Tuesday

In the United States Electoral System, Super Tuesday refers to the Tuesday in February or March of a presidential election year. During this period, majority of the states conduct the primary elections to select their delegates to national conventions where presidential candidates for each party are officially nominated. The Super Tuesday 2012 was held on March 6 in States like; Alaska, Georgia, Idaho, Massachusetts, North Dakota, Ohio, Oklahoma, Tennessee, Vermont and Virginia. The Super Tuesday tweets collected for the experiments considered the official hashtags #*SuperTuesday* and other main hashtag keywords such as the four main Republican candidates namely, Mitt Romney, Ron Paul, Newt Gingrich, and Rick Santorum as well as the ten states and the major newsagents reporting the events.

TRCM system is trained to discover *ARs* present in tweets hashtags of selected datasets. We map hashtag keywords contained in the *ARs* obtained to related real-life topics provided by Aiello et al [Aiello et al., 2013].

5.6.2 Data Collection and Preprocessing

We use a collection of tweets relating to the 3 topics (FA cup final 2012, US elections 2012 and Super Tuesday 2012). These collections include main keywords which relates to each topic. We extract and analyse tweets that include hashtags with their timestamps as shown in Fig. 5.2. The timestamps enable us to map the time-slot of hashtag keywords detected by *TRCM* with the unfolding of the respective events in the ground truth. The FA Cup collection has 444,291 tweets over a period of 72 hours (4 - 6 May 2012), however, we analyse only 50.6% (224,291) of the total collection. This percentage represents the number of tweets posted on-line during the game (May 5 2012, 5:15pm to 7:00pm). For the US Election 2012 and the Super Tuesday, there were collections of 3,837,291 and 474,109 respectively.

We divide the English FA Cup Finals 2012 tweets into about 2,000 tweets/time slot and 1 minute update rate due to the rapid evolvement rate of game (where 2 goals can be scored within 2 minutes). For the US Presidential Election 2012, we divide the tweets into 20,000 tweets/time slot and 10 minutes update rate. For the Super Tuesday divide the tweets into 10,000 tweets/time slot and 1 hour update rate because events in political datasets were discovered to evolve less frequently and span over a longer period of time.

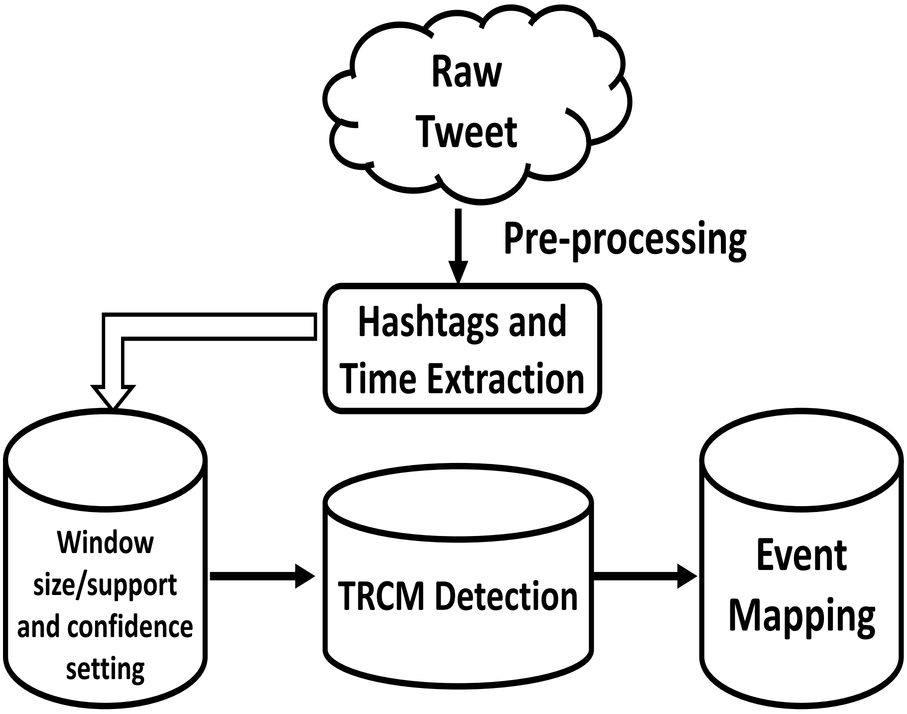
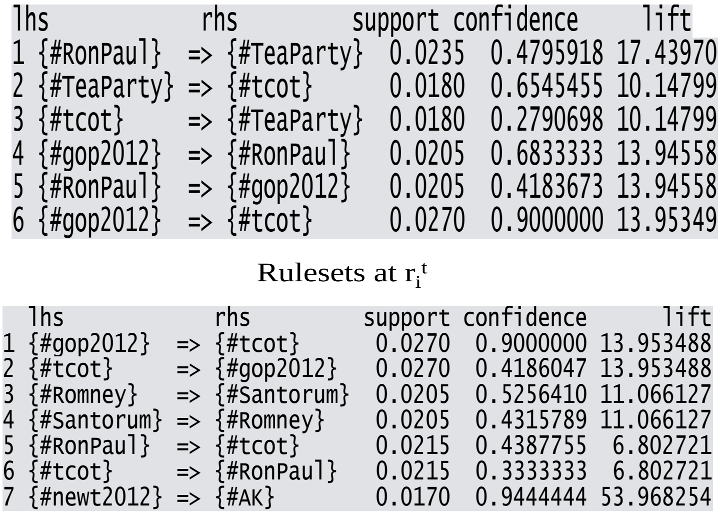

Raw Tweet
Pre-processing
Hashtags and Time Extraction
Window size/support and confidence setting
TRCM Detection
Event Mapping

FIGURE 5.2: Event Detection Process

5.7 Experimental Setup

We set out to conduct *TDT* experiments that automatically detect real-life topics from hashtags using *ARM*. To achieve this, we divide tweets in each of the datasets into window size and specify their update rate as explained in Section 5.6.2. For the English FA Cup final we select 1 minute update period, for the US Presidential election and Super Tuesday we select 10 minutes and 1 hour respectively as in [Aiello et al., 2013]. These settings were found to yield better results on the datasets after empirical fine tuning. The different settings enhance the precision of rules returned by *TRCM* within each time-slot. We also set both the support and confidence to 0.001 after carrying out preliminary study to confirm the setting that best optimises *TRCM's* efficiency on the case study datasets. In these experiments, we are more concerned with extracting as many relevant hashtag keywords that are related to targeted events (precision) as possible.

We extract all hashtags in the tweets and defined a function that finds matching terms in *lhs* and *rhs* (left-hand side and right-hand side) rules of r_i^t and at r_j^{t+1}. These are used to set the *TRCM* rules. We find matching values in *lhs* and *rhs* of $r_j^{t+1}$and r_i^t as presented in Fig. 5.3 where *#tcot* and *#RonPaul* are unexpected consequent rules (similar *lhs* but different *rhs*), where *#Romney* is a new rule (no matching found). *TRCM* is identified by defining the thp_{ij} and thq_{ij} (left hand side and right hand side user-defined threshold (RMT)) which are set between 0 and 1 with equal value for both threshold. The experiments were conducted in RStudio Version 3.0.0 (2013-04-03), Platform: $x86\_64 - w64 - mingw32/x64$ (64-bit) and processed on Windows 7 Enterprise of 8.00 RAM memory size and CPU @ $3.20GHz$.



```
lhs            rhs          support confidence     lift
1 {#RonPaul}  => {#TeaParty}  0.0235  0.4795918 17.43970
2 {#TeaParty} => {#tcot}      0.0180  0.6545455 10.14799
3 {#tcot}     => {#TeaParty}  0.0180  0.2790698 10.14799
4 {#gop2012}  => {#RonPaul}   0.0205  0.6833333 13.94558
5 {#RonPaul}  => {#gop2012}   0.0205  0.4183673 13.94558
6 {#gop2012}  => {#tcot}      0.0270  0.9000000 13.95349
```

Rulesets at r_i^t

```
  lhs            rhs          support confidence      lift
1 {#gop2012}  => {#tcot}       0.0270  0.9000000 13.953488
2 {#tcot}     => {#gop2012}    0.0270  0.4186047 13.953488
3 {#Romney}   => {#Santorum}   0.0205  0.5256410 11.066127
4 {#Santorum} => {#Romney}     0.0205  0.4315789 11.066127
5 {#RonPaul}  => {#tcot}       0.0215  0.4387755  6.802721
6 {#tcot}     => {#RonPaul}    0.0215  0.3333333  6.802721
7 {#newt2012} => {#AK}         0.0170  0.9444444 53.968254
```

Rulesets at r_j^{t+1}

FIGURE 5.3: Rules Matching for Super Tuesday

Events mapped by our system in the sports dataset include goals, bookings, substitutions, shot-on-targets, free kicks and foul plays. In the US Election 2012, events detected include the disruption of the two candidates' election campaign by Hurricane Sandy, California death penalty ban rejected by voters, the presidential election result and Obama's victory speech. In the Super Tuesday, events detected include the election results in different states of the United States. For the experiments, an item h is any hashtag present in the tweet, while the transaction is the tweet message that occurs in a time slot T. The number of times that any given set of hashtags occur in the time slot is referred to as its **support**, and hashtags that meet a minimum support is referred to as a **frequent pattern**. To confirm event detection in the datasets, we examine hashtag keywords present in the *ARs* returned within each time-slot and rank them at 3 levels. First, we analyse hashtag keywords under the *unexpected consequent/ unexpected conditional rules*, then those under the *emerging rules*. Finally, we combine both the *unexpected* and the *emerging* rules. All the hashtags detected in each time-slot at the 3 levels were recorded along with the time the tweets were posted on Twitter to evaluate our system's performance. We establish a match if the returned hashtags in each time-slot contain at least one of the key terms used in the ground truth within the same time frame the detection occurred as shown in Tables. 5.1 and 5.2. We confirmed that the hashtag keywords detected as unexpected and emerging rules were those that best represent different event highlights in the datasets when mapped to the ground truth.

5.8 Experimental Results

To validate our topic detection technique we used ground truth from Main stream Media (MSM) previously used for annotation in [Aiello et al., 2013] for the 2 political datasets (that is, US Presidential Election 2012 and Super

Table 5.1: Samples of Mapped Rules in US Election 2012 dataset

Time Frame	News Samples	Hashtags Samples
7.11.12/04.04 - 04:10	Clinton 2016? Hillary is top choice as Democrats turn to next election: The guardian	#*Hillary*2016,#Election
7.6.11.12/04:15 – 04:20	2012 Presidential Election: Hurricane Sandy Alters Romney, Obama Campaign Plans: ABC News	#*Sandy*,#election
7.11.12/08.57 – 09:05	Election Results 2012: President Barack Obama's Family Calls White House Home for Four More Years – ABC News	#*Obama*#Fourmoreyears
7.11.12/1:45 - 1:55	Election 2012: Obama's Complete Victory Speech (The New York Times)	#*Victoryspeech*,#Obama

Table 5.2: Samples of Mapped Rules in Super Tuesday 2012 dataset

Time Frame	News Samples	Hashtags Samples
4.3.12/10:24 - 11:24	Romney wins Washington state caucuses (CNN: 4 March 2012, 22:15 GMT)	#Mitt2012 #Romney
4.3.12/13:38 - 14:12	Super Tuesday: Romney edges out Santorum Ohio – (BBC: 7 March 2012, 09:20am)	#Santorum #*Supertuesday* #Romney
4.3.12/10:24 - 11:24	Ron Paul was Super Tuesday's big winner (The Wire - 7 March 2012: 11:08am ET)	#RonPaul
4.3.12/16:15 - 16:35	Romney humbled as Santorum roars back into Republican race (The Time - 7 Feb. 2012)	#Romney #Santorm
4.3.12/16:35 - 16:57	Why Ron Paul matters more than Newt Gingrich (The Washington Times - 9 April 2012)	#RonPaul #Newt
4.3.12/17:45 - 18:05	For Gingrich, Georgia is must on Super Tuesday (CNBC - 2 March 2012)	#Supertuesday #Gingrich #Georgia
7.3.12/19:45 - 20:45	For Gingrich wins Georgia (The Guardian - 7 March 2012)	#Supertuesday #Gingrich #Georgia

Tuesday). For the sports dataset we generate ground truth from the BBC sports official website [2]. The FA Cup final match between *Chelsea Football Club* and *Liverpool Football Club* produce event highlights that were detected by *TRCM*. Our system was able to detect events such as **goals scored, bookings, player substitutions, free kicks, offside, misses, saves and clearances**. Event detection mapping was carried out manually. For the US Elections 2012 our system mapped 11 out of 24 topics in the ground truth (45%). We show samples of our system detection in Table. 5.3. The effectiveness of measure for our system is discussed in Section 5.9

Table 5.3: Table Showing TRCM Event Detection for FA Cup 2012 Dataset

	G	Sub	BK	FK	S	CL	OS	BL	MS	TE
Ground Truth	3	4	3	10	11	19	4	3	2	59
TRCM	3	3	1	10	9	19	4	3	2	54

In Table.1 GT = Ground Truth; G = Goals; Sub = Substitutions; BK = Bookings; FK = Free kicks; CL = Clearances; OS = Offsides; BL = Blocks; MS = Misses; TE = Total Event

5.9 Effectiveness Measure

Recall and precision are performance measurement metrics used in Information Retrieval (IR) to measure the performance of a system. Precision is the percentage of relevant instances identified by the system, while recall is the percentage of relevant instance classified correctly [Baeza-Yates et al., 1999]. The system error rates are used to evaluate appropriateness of the system. Other single-valued measures have been implemented [van Oorschot et al., 2012] however,

[2] http://www.bbc.co.uk/sport/0/football/17878435

F-Measure, which is a a mixture of precision and recall, turn out to be the most dominant approach for text classification evaluation.

In this experiment, we measured the performance of our system by applying **precision, recall** and **F-Measure** to the three datasets. In summary, we classified all hashtag keywords identified as unexpected and emerging rules in each of the datasets and map them to the ground truth at three levels of **Performance Variation (PV)**. First, we mapped **unexpected rules only** keywords, then we mapped **emerging rules only** and lastly, we mapped the combination of both unexpected and emerging rules as shown in Tables 5.4,5.5, 5.6 and Fig. 5.4. This was carried out to demonstrate the effectiveness of each *TRCM* rules at different mapping levels. The *PV* shows that the application of both unexpected and emerging rules on the datasets enhanced the performance of our system particularly on the sports dataset.

Table 5.4: Table showing Precision PV

Dataset	Unexpected	Emerging	Both
FA Cup	91.5%	5.8%	96.6%
US Election	34.6%	19.2%	53.8%
Super Tuesday	37.5%	25%	62.5%

Table 5.5: Table showing Recall PV

Dataset	Unexpected	Emerging	Both
FA Cup	85.7%	3.89%	64.0%
US Election	40.9%	20.8	70.0%
Super Tuesday	20%	14.2%	55.5%

Table 5.6: Table showing F-Measure PV

Dataset	Unexpected	Emerging	Both
FA Cup	88.5%	4.40%	76.9%
US Election	37.4%	19.96%	60.0%
Super Tuesday	26%	18.1%	58.79%

The calculations for the precision, recall and F-Measure effectiveness measures is expressed in Table 5.7.

Table 5.7: Table showing the effectiveness measure of TRCM

	Relevant	Not-relevant
Retrieved	A = 54	B = 9
Not retrieved	C = 5	-

For the purpose of illustration, the retrieved hashtags in the table are those that our system classified as positive instances in the football game, and the relevant hashtags are those that we manually judged relevant to the game. The calculations were replicated for the 2 political datasets used in the experiments. The effectiveness measure of our system implies that at least one out of the hashtags detected must be present in the relevant time-slots in the ground truth.

$$Precision = P = \frac{A}{A+C} \times 100 = 91.5\%$$

$$Recall = R = \frac{A}{A+B} \times 100 = 85.7\%$$

$$F\text{-}Measure = F = \frac{2PR}{P+R} = 88.5\%$$

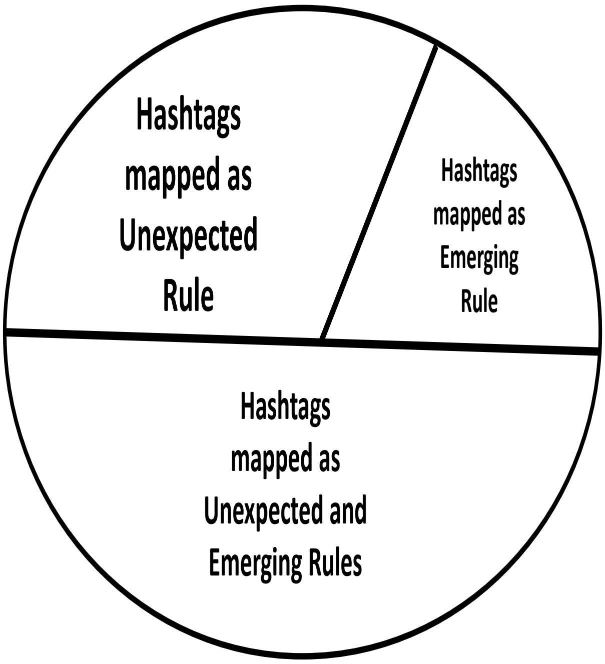

Hashtags mapped as Unexpected Rule
Hashtags mapped as Emerging Rule
Hashtags mapped as Unexpected and Emerging Rules

FIGURE 5.4: Illustration of the Performance Variation

5.10 Performance Analysis

The experiments conducted in this chapter revealed that *TRCM* performed better on dataset from the sports domain. This can be attributed to the short timeline and rapid evolvement of highlights for sporting events (90 minutes to 120 minutes in the case of football game). On the other hand, events in politics are known to have longer timeline making event/topic detection and tracking more complex. In addition, the ground truth we used for the FA Cup 2012 presents one topic per time slot whereas topics in the political datasets occurred in parallel. While the sports event ground truth covers all the highlights of the game, those for the two political datasets did not capture some of the important events that occurred during the US Elections 2012 and the Super Tuesday 2012. Further investigations into the hashtag keywords classified as *false positives* according to the ground truth, are found as news headlines on the websites of other newsagents. An example of such headlines is the one reported on CNN websites under the title, "**California Proposition 34: Ban Death Penalty**" and Huffington Post of November 7, 2012 under the title, "**California Death Penalty Ban Rejected By Voters**". We also consider Wikipeda databases for US Elections and Super Tuesday 2012, during the US elections 2012, Wikipedia collected and posted online, all the elections results displayed in texts, tables and graphs, making it easy to retrieve vital and credible information on the events. With these findings, we established that if we consider multi ground truth for our analysis, *TRCM* will exhibit enhanced results across all the three performance measures used namely; **precision, recall and F-Measure** For the political datasets performance measure enhancement of up to 30% can be achieved.

5.11 Summary

The enormous data generated on Twitter network requires data mining techniques such as *ARM* to analyse tweets for necessary use by different entities. Detection, extraction and presentation of tweets' hashtags related to specific event/topic in real-life express the relevance of tweets' hashtags for *TDT* purpose. As topics/events in real-life undergo different phases so do tweets and hashtags included in tweets. The experiment reported in this chapter validated the *TRCM* method that detects rules changes based on hashtags present in tweets and how the changes relate to events/occurrences in real-life scenarios. The experiments demonstrate the relevance of tweets' hashtags where appropriately applied. This also underpin the fact that Twitter is becoming more of information network than just a social network. This can be justified by *TRCM* experimental results which detect event highlights in the three datasets analysed. All the rules detected in these experiments are applied to related real-life situations and can be adopted as decision support tool for different entities, including individuals, organisations and government. In Chapter 6 we shall conduct qualitative studies of *TRCM* by tracking a number of real-life topics/events that were widely publicised in real-life news at different periods of time. We shall be adopting a novel methodology built from *TRCM* termed *RTI-Mapping* to map and track evolvements of related news using hashtag keywords detected in *ARs* present in tweets.

Chapter 6

A Qualitative Study of *TRCM* for Event Tracking

The findings reported in this chapter have been published in [1]. The outcomes of the experiments conducted and the qualitative case studies analysed in the book chapter are presented in this chapter and dully referenced. Furthermore, we propose a novel visualisation method termed **TRCM-viz**. The method was built with a graph-based **NoSQL** database system namely **Neo4j**. As an example of presenting the method, we apply it to one of the experimental case studies (#*Boston*) dataset to show the applicability of the method.

[1] Adedoyin-Olowe, M., Gaber, M. M., Stahl, F., & Gomes, J. B. (2015). Autonomic Discovery of News Evolvement in Twitter. In Big Data in Complex Systems (pp. 205-229). Springer International Publishing

6.1 Introduction

In chapter 5 we validated quantitatively our research methodology named *TRCM* on three real datasets from two diverse domains. We evaluated the performance of *TRCM* on the datasets by applying precision, recall and F-Measure metrics. In this chapter, we conduct qualitative studies of *TRCM* by tracking a number of real-life topics/events that made news headlines at specific periods of time. We adopt RTI-*Mapping* (Rule Type Identification-Mapping) built from *TRCM*. We used the method to map hashtags detected in the experiment to real-life news broadcast by traditional newsagents. We track these hashtags in news updated within a specified period of time as the news unfolds in real-life. We then visualise the hashtags and *ARs* detected in one of the experimental case studies using our novel visualisation method termed **TRCM-Viz** built with **NoSQL Neo4j**. We cluster the hashtags and rules to view interesting hashtags and the rules they evolved over. This allows for ease of analysis.

6.2 Hashtag Evolvements in Tweets

It has been observed that users replace hashtags used in tweets of specific event based on the evolvement of the event in real-life. Tweets about a local event may trigger a global discussion thus resulting in hashtag modifications. An example of this is the news of 'baby59' found in a sewage pipe in China on May 25, 2013 [2]. The news started locally, and later became global news. Hashtags used in related tweets when the news first unfolded changed as information regarding the occurrence continued to evolve. However, this is not always the case as Twitter users are sometimes not quick to update hashtags used in tweets, even

[2] http://www.theguardian.com/world/video/2013/may/30/china-landlady-finding-baby-59-sewage-pipe-video

when the content of their tweet changes.

We developed RTI-*Mapping* (Rule Type Identification-Mapping) from *TRCM* primarily to achieve the following:

- Map evolving *ARs* with related evolving news in real-life.
- Track updates of evolving news as broadcast by traditional newsagents.

6.3 Which comes first, the News or the Hashtag? - The "TwO - NwO" State

Twitter users oftentimes tweet about events in real-time or disseminate on-the-spot information on the network. Such tweets may trigger the broadcast of the events/information by newsagents as presented in Fig. 6.1. An example of such situation is the news of the death of American female pop star, Whitney Houston which was posted on Twitter before its broadcast on news media [Whiting et al., 2012] as breaking news. In this case the tweet comes before the news **tweet-originated**. On the other hand, an event or topic broadcast by newsagents may result in Twitter users hashtagging keywords in the news while expressing their opinion/sentiment on the topic via Twitter network. We refer to such topic as **news-originated**. The opinion/sentiment expressed on a topic (tweets-originated or news originated) may go on to result in chains of news updates to earlier news reports. This is termed *"TwO - NwO"* state in the context of this thesis.

Topic/event posted on Twitter before its broadcast by newsagents in real-life is termed ***TW*****eet** ***O*****riginated topic or** ***TwO*** topic. While those topics/events broadcast by newsagents before they are being posted on Twitter network are termed ***N*****eWs** ***O*****riginated or** ***NwO*** topics. Apart from these two states, a planned event can be tweeted using what is known as the *official hashtag* for the purpose of publicity/awareness. This is referred to as ***Ev*ent *O*riginated topic or *EvO*** topic.

In Fig. 6.1 the first sequence demonstrates an occurrence (for example the Boston marathon bomb blast in 2013) which resulted in *#BostonMarathon* and *#BostonBombBlast* being included in tweets pertaining to the incident *(event → hashtag → news)*. The second sequence can be used to demonstrate the broadcast of a planned event televised or reported by traditional news media before being tweeted on-line. Another illustration for *'Which come first'* was revealed during the tracking of events evolvements in the Business news case study as presented in Fig. 6.8. It would be observed that some of the news that are related to *#BusinessNews* were already being reported by traditional newsagents before RTI-*Mapping* experiment was conducted with the use of the *#BusinessNews* keyword. This implies that, the *news comes first (NwO)*.

The "TwO - NwO" state demonstrates how topic/event tweets and related real-life news broadcast by traditional newsagents can be interrelated and subsequently compared to corroborate the authenticity of Twitter posts. However, while traditional newsagents refine their news, Twitter community are too quick to post unverified contents on-line, which could be misleading. Such situation is explained in trend analysis *TA* and time frame windows *TFWs* of rule evolvements discussed in Section 4.7.

6.4 Tracking Rule Evolvement in Tweets

RTI-*Mapping* offers an autonomic method of detecting evolving *ARs* present in tweets of related real-life events and tracking the evolvements of the news when updated by traditional newsagents. The experimental workflow is built to allow Twitter users track evolving news, and news updates from within Twitter network. Events in real-life are often on going and requires newsagents to broadcast timely updates of trending news. Newsagents update news of

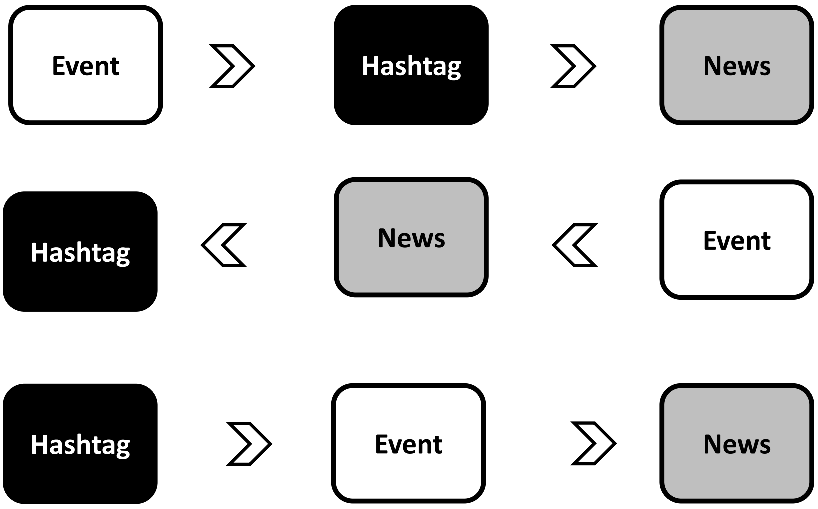


FIGURE 6.1: Which Comes First?

events/occurrences to keep their audience abreast of how events unfold. RTI-*Mapping* keep users up-to-date with the news updates in a quick, effective and efficient way.

We apply the p_{ij} and q_{ij} equations presented in Chapter 4 to extract the similarities in the conditional and consequent parts of rule i and rule j at time t and $t + 1$, respectively. However, unlike the approach taken in [Song et al., 2001] as discussed in Chapter 4, we adopt the transactional view rather than the relational view. Thus, our equations are simpler and faster to compute. More importantly, the transactional view is a better fit for hashtags in Twitter, as existence or absence of the hashtag is what contributes to forming the association rules.

Notation used for RTI-*Mapping* algorithm is presented in Table 6.1.

Table 6.1: Notation for Algorithm 3

$t0$	tweets retrieved at time t
$t1$	tweets retrieved at time $t+1$
$r0$	rules obtained from $t0$
$r1$	rules obtained from $t1$
$lhsT$	left hand rule threshold
$lhsMax$	rules similar in left hand of $R1$ and $R2$
$rhsT$	right hand rule threshold
$rhsMax$	rules similar in right hand of $R1$ and $R2$
$ruleSimLH$	degree of similarity in the left hand
$ruleSimRH$	degree of similarity in the right hand

Algorithm 3 is designed to detect chains of rule evolvements from t to $t1$, $t2$...tth in a consecutive way. The threshold of the rules similarities in the consequent and conditional part of rulesets at t and $t+1$ are represented with $lhsT$ and $rhsT$. We find similar patterns of rules, *r0* and *r1* and then inspect *r1* using $TRCM$ to detect the rule change evolvement at $t+1$. Rule patterns in $r1$ show the rule trend over the period of evolvement *(r0 to r1)*. We relate the $TRCM$ rule trend of tweets to real events in Section 6.6.

Algorithm 3 TRCM-RTI Algorithm

```
  Function findPattern(newRule, ruleSet)
Require: lhsT and rhsT
Ensure: lhsMax = 0 and rhsMax = 0
  for all oldRule in ruleSet do
    lhsMax = max(lhsMax, ruleSimLH(newRule,oldRule))
    rhsMax = max(rhsMax, ruleSimRH(newRule,oldRule))
    if $(lhsMax \geq lhsT \textbf{and} rhsMax \geq rhsT)$ then
      return $Emerging$
    end if
  end for
  if $(lhsMax \geq lhsT)$ then
    return $UnexpectedConsequent$
  else
    if $(rhsMax \geq rhsT)$ then
      return $UnexpectedConditional$
    end if
  end if
  return $New$
  EndFunction
Require: $r0$, $r1$ rules mined from the tweets retrieved at time $t$ and $t+1$
  for all $newRule$ in $r1$ do
    $newRule.TRCM$ = findPattern(newRule,r0)
  end for
```

6.5 Methodology

6.5.1 Data Collection

Unlike the datasets used for the experiments in Chapter 5, the datasets analysed for event tracking in this chapter were obtained by crawling Twitter network using its *API* within a specific time frame. We collected a small sample of about 100 tweets per dataset. Datasets analysed were from different domains such as **politics** (#*MargretThatcher*), **social,** (#*GayMarriage*), **socioeconomic** (#*Shutdown*) and **business** (#*BusinessNews*).

6.5.2 Experimental Setup

The experiments conducted in Chapter 5 detected highlights of real-life events while the experiments in this chapter track widely publicised real events over a specific period. We used Twitter *API* to crawl Twitter using carefully chosen hashtag keywords that best describe the targeted event for analysis. We trained the algorithm to extract tweets with the specified hashtags over a consecutive period of days (mostly four days interval for each of the events analysed). The algorithm then extract hashtag keywords present at the two time period and obtain *ARs* in $r0$ and $r1$ at t and $t+1$ respectively using *Apriori*. We set the support and confidence parameters to 0.01 (support) and 0.05 (confidence) after preliminary testing of rule setting that best return relevant rules that can be mapped to real-life news. The chosen setting is found to increase the performance accuracy of RTI-*Mapping*. Hashtags in both *left hand side* and *right hand side* of rules at time t and $t+1$ *(lhsT and rhsT)* are matched to detect *TRCM* at time $t+1$ using RMT $\in$ [0,1].
Where rule similarity is detected in the *lhs* (left hand side/conditional part) of $r1$ *(sim_lhs* $\geq$ *lhsT)*, then unexpected consequent rule evolvement is said to have occurred. If similarity is detected in the *rhs* (right hand side/consequent part) of $r1$ *(sim_rhs* $\geq$ *rhsT)*, then unexpected conditional rule evolvement has occurred. However, it should be noted that unexpected consequent and unexpected conditional rule evolvements are the same in real-life situation and are therefore treated as one evolving rule pattern. On the other hand, emerging rule evolvements occur when there is similarity in both the *lhs* and *rhs* of rules *(sim_lhs* $\geq$ *lhsT& sim_rhs* $\geq$ *rhsT)*. All the *ARs* detected in tweets are mapped to traditional news in real-life and tracked over specific time as different scenarios of the event unfolded as shown in Fig. 6.2.

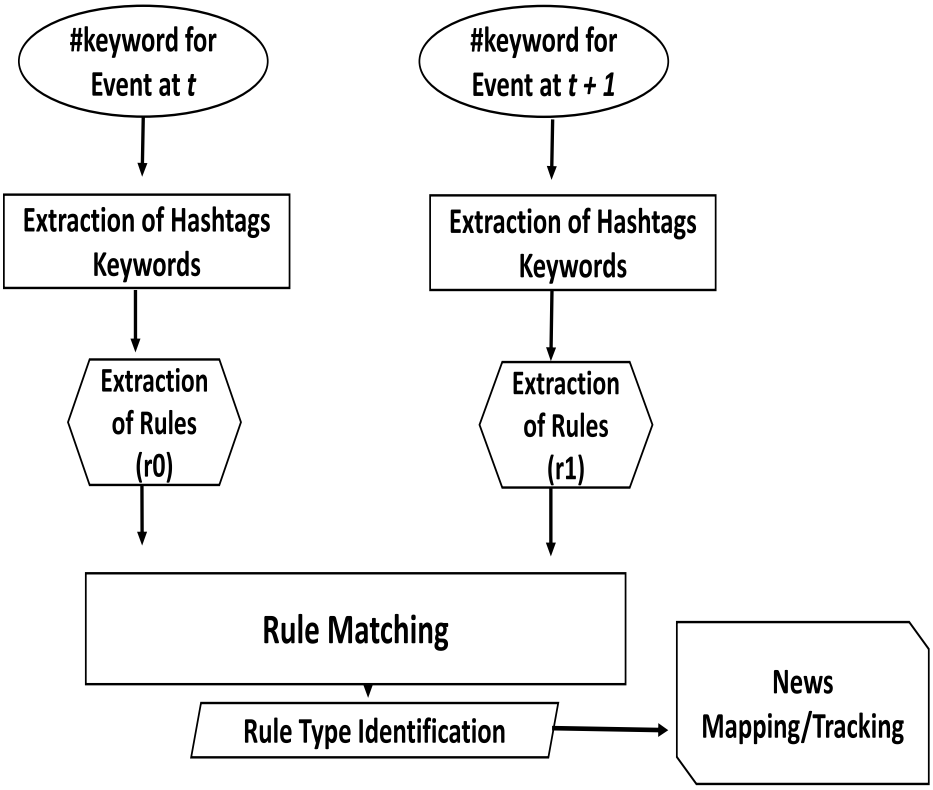

#keyword for Event at *t*
#keyword for Event at *t + 1*
Extraction of Hashtags Keywords
Extraction of Hashtags Keywords
Extraction of Rules (r0)
Extraction of Rules (r1)
Rule Matching
Rule Type Identification
News Mapping/Tracking

FIGURE 6.2: RTI-Mapping Process

6.5.3 Experimental Results

Results for mapping the evolving *ARs* in tweets with news emergence and news updates of traditional newsagents were completed manually. *ARs* in $t + 1$ classified into unexpected consequent, unexpected conditional and emerging rules are mapped as required. For each evolving *AR* detected in each case study, all the hashtag keywords retrieved within the detected *ARs* are used as search terms in on-line news databases of renown newsagents used as ground truth for validation of our system. RTI-*Mapping* revealed hashtags detected under emerging rules as mostly breaking news. This explains the rapid evolvement of such hashtags in related tweets. The four case studies in subsequent sections are used to evaluate our system.

Emerging rules and unexpected rules are considered in this experiment because they are evolving *ARs* that best describe the dynamics of real-life news and news updates. As defined in Section 4.4, all rules in $t + 1$ are new until a matching is found in tweets at t. For this reason, new rules are not analysed in all the experiments conducted in this research.

6.6 Experimental Case Studies

6.6.1 Case Study 1 - *#MargretThatcher*

The news of Margaret Thatcher's (former Prime Minister of Great Britain) death on the 8^{th} of April 2013 was widely reported as breaking news all around Europe and in other parts of the world. Tweets about her death were posted on Twitter resulting in the use of *#MargretThatcher* and *#Thatcher*. *TRCM* experimental study conducted on these two hashtags within 25 days of her death revealed different patterns of rule evolvements of hashtags used in tweets. Hashatags identified by *TRCM* in the first few days of Thatcher's death were mostly emerging rules, example of such hashatags include *#MargretThatcher*, *#Thatcher*, *#RIP*, *#BBC* and *#IronLady*. However, by the 21^{st} day, some of the rules (such as *#RIP* and *#BBC*) were dead implying that they had less than 21 days lifespan on Twitter. *#Funeral* ⇒ *#Thatcher* evolved from *emerging* to *Unexpected Conditional* and *Unexpected Consequent (TFWs)*. The evolvements were mapped to widely publicised controversy on the £10 million state money spent on her burial which some individuals and groups consider as wasteful spending. News regarding the funeral spending controversy was reported by different newsagents at the time [3], [4], [5]. The hashtags were mapped to headlines reported in the on-line version of "**The guardian**" newspaper of Tuesday 16 April 2013 at 14.45 BST captioned "**Margaret Thatcher's funeral: 23 things you could pay for with £10m**" (as shown in Fig. 6.4). The *Unexpected Conditional* evolvement of $\#Unions \Rightarrow \#Thatcher$ was mapped to workers' day celebration on May 1, which brought back discussions on Thatcher's presumed adverse policy on workers' union during her time in office. On the other hand, the *Unexpected Conditional* rule evolvements of $\#AcidParty \Rightarrow \#Thatcher$ was mapped to Thatcher's negative views on Acid party and hardcore music being embraced by young British in the 80s. Rules such as $\#Channel4 \Rightarrow \#RandomActs, \#RandomActs \Rightarrow \#Thatcher, \#Channel4 \Rightarrow \#AcidParty, \#RandomActs \Rightarrow \#AcidParty$ became common because of Channel 4's (UK public-service television broadcast) introduction of a peculiar Acid party song on their 'Random Acts' programme. Margret Thatcher's speeches were randomly chosen and composed into an 'Acid party' song which was uploaded on YouTube. All these *ARs* were tracked to $\#Thatcher$ and $\#MargretThatcher$ in our experiment. Although $\#Channel4 \Rightarrow \#RandomActs, \#RandomActs \Rightarrow \#Thatcher, \#Channel4 \Rightarrow \#AcidParty, \#RandomActs \Rightarrow \#AcidParty$ do not imply $\#Thatcher$ or $\#MargretThatcher$ their origin can be easily traced to $\#Thatcher$ and $\#MargretThatcher$ *(rule trace)*. An example of such rule is $\#channel4 \Rightarrow \#RandomActs$. We present a "**rule chain**" demonstrating the evolvement of rules in $\#MargretThatcher$ in Fig. 6.3.

[3] http://www.theguardian.com/news/datablog/2013/apr/16/margaret-thatcher-funeral-10-million

[4] http://www.independent.co.uk/news/uk/home-news/bishop-rt-rev-tim-ellis-warns-that-10m-costof-margaret-thatchers-funeral-is-asking-for-trouble-8572587.html

[5] http://www.mirror.co.uk/news/uk-news/margaret-thatcher-funeral-forget-8million-1820189

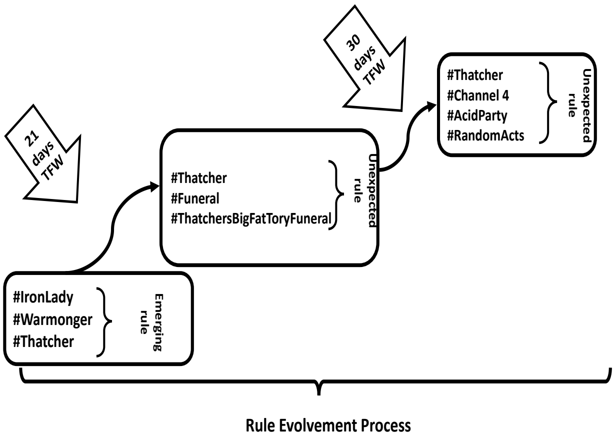


FIGURE 6.3: Rule Chain for #*MargretThatcher*

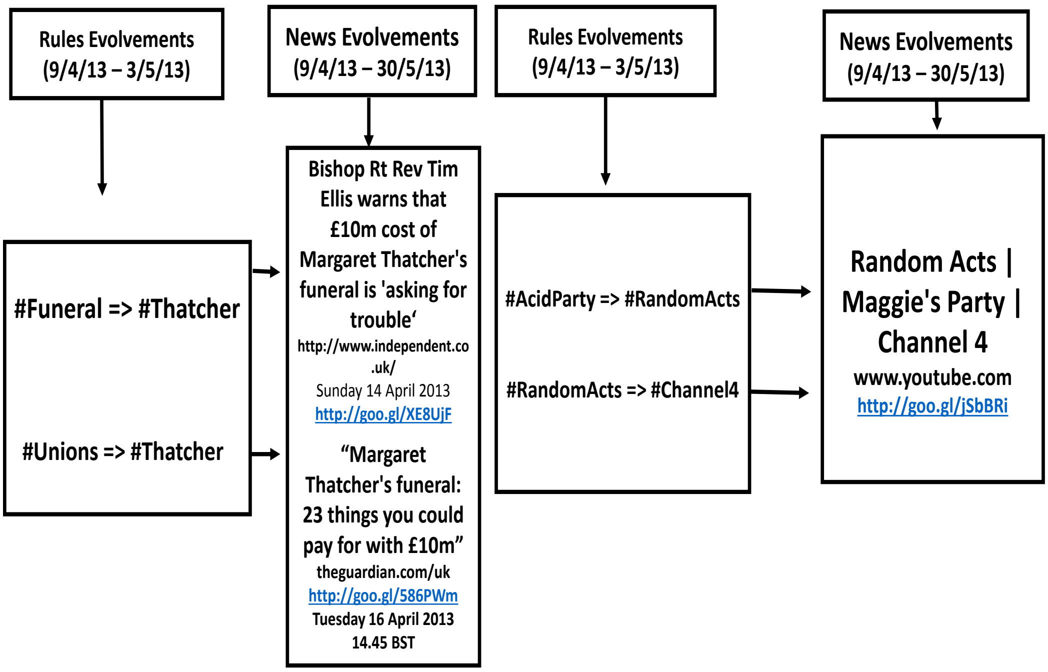

Rules Evolvements (9/4/13 – 3/5/13)
News Evolvements (9/4/13 – 30/5/13)
Rules Evolvements (9/4/13 – 3/5/13)
News Evolvements (9/4/13 – 30/5/13)
#Funeral => #Thatcher
#Unions => #Thatcher
Bishop Rt Rev Tim Ellis warns that £10m cost of Margaret Thatcher's funeral is 'asking for trouble'
http://www.independent.co.uk/
Sunday 14 April 2013
http://goo.gl/XE8UjF
"Margaret Thatcher's funeral: 23 things you could pay for with £10m"
theguardian.com/uk
http://goo.gl/586PWm
Tuesday 16 April 2013 14.45 BST
#AcidParty => #RandomActs
#RandomActs => #Channel4
Random Acts | Maggie's Party | Channel 4
www.youtube.com
http://goo.gl/jSbBRi

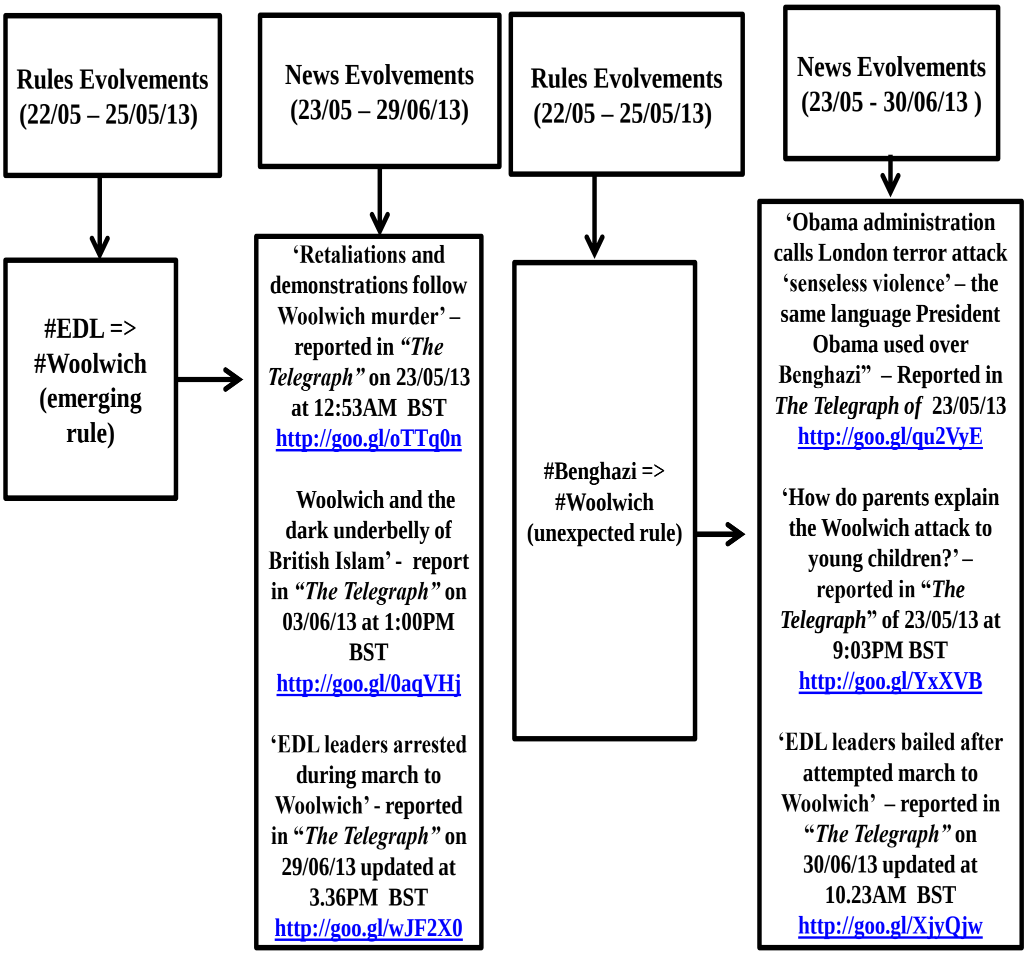
FIGURE 6.4: Rule Mapping for #*MargretThatcher*

6.6.2 Case Study 2 - *#Woolwich*

The murder of a British soldier in Woolwich, South-East of London on May 22, 2013 made global news while Twitter users labelled tweets that are related to the occurrence with different hashtags. *#Woolwich* was used as keyword to crawl Twitter using its *API*. The experiment was conducted within 7 days of the occurrence. *#EDL* ⇒#Woolwich and *#Benghazi* ⇒#Woolwich evolved as *Emerging* and *Unexpected Consequent* rules respectively. The hashtag keywords identified in the rules were used to search on-line version of "**The Telegraph**" newspaper. *ARs* detected between May 22, 2013 and May 25, 2013 were mapped to news headlines and news updates of "The Telegraph" between the May 23, 2013 and June 30, 2013 as presented in Fig. 6.5. The dates of the tweets and those of the news reports in real-life demonstrates the "NwO, TwO concept" which explains which comes first, whether the tweets containing the hashtag keywords identified by *TRCM* or their related news in real-life.

#EDL ⇒#Woolwich evolved as *emerging rule* between May 22 and May 25, 2013. The rule was mapped to news headlines in "The Telegraph" during and after the period when the rule evolved on Twitter network. The first news item mapped to *#EDL* ⇒#Woolwich was dated May 23, 2013 at 12:53 am BST with headlines "**Retaliations and Demonstrations follow Woolwich Murder**". Another update of the news was mapped to the headlines captioned "**Woolwich and the dark underbelly of British Islam**" on June 3, 2013. By the June 29, 2013 another update was mapped with caption "**EDL Leaders Arrested during March to Woolwich**".

On the other hand, *#Benghazi* ⇒#Woolwich evolved as *unexpected rule*. *#Benghazi* and *#Woolwich* were used as search terms in the news headlines of "**The Telegraph**". The first news that was related to *#Benghazi* ⇒#Woolwich was mapped on May 23, 2013 with caption "**Obama Administration Calls London Terror Attack 'Senseless Violence' the Same Language President Obama used over Benghazi**". Update on *#Woolwich* was tracked the next day (24 May 2013) at 12:00PM BST in the "**The Telegraph**" with the caption "**How do parents explain the Woolwich attack to young children?**". The last mapping was completed on June 30, 2013 at 16:23AM BST with headlines captioned "**EDL leaders bailed after attempted march to Woolwich**".


Rules Evolvements (22/05 – 25/05/13)
#EDL => #Woolwich (emerging rule)
News Evolvements (23/05 – 29/06/13)
Rules Evolvements (22/05 – 25/05/13)
#Benghazi => #Woolwich (unexpected rule)
News Evolvements (23/05 - 30/06/13)

'Retaliations and demonstrations follow Woolwich murder' – reported in *"The Telegraph"* on 23/05/13 at 12:53AM BST http://goo.gl/oTTq0n

Woolwich and the dark underbelly of British Islam' - report in *"The Telegraph"* on 03/06/13 at 1:00PM BST http://goo.gl/0aqVHj

'EDL leaders arrested during march to Woolwich' - reported in "*The Telegraph"* on 29/06/13 updated at 3.36PM BST http://goo.gl/wJF2X0

'Obama administration calls London terror attack 'senseless violence' – the same language President Obama used over Benghazi" – Reported in *The Telegraph of* 23/05/13 http://goo.gl/qu2VyE

'How do parents explain the Woolwich attack to young children?' – reported in "*The Telegraph*" of 23/05/13 at 9:03PM BST http://goo.gl/YxXVB

'EDL leaders bailed after attempted march to Woolwich' – reported in "*The Telegraph"* on 30/06/13 updated at 10.23AM BST http://goo.gl/XjyQjw

FIGURE 6.5: Rule Mapping for #*Woolwich*

6.6.3 Case Study 3 - *#GayMarriage*

Gay marriage debates became very intense in many countries of the world in 2013. Religious bodies, groups and individuals expressed their views on the passing of the bill legalising gay marriage in some countries in Europe and America. While many African countries rebuff legislation of gay marriage, the governments' bill covering England and Wales was passed in the House of Commons in 2013. RTI-*Mapping* experiments conducted on *#gaymarriage* between June 1 and June 4, 2013 revealed a number of evolving *ARs*. *#gayrights* ⇒ *#gaymarriage* and *#politicalparties* ⇒ *#gaymarriage* evolved as *unexpected rules*, the rules were mapped to "**The BBC Politics on-line News**" from June 4 to June 27, 2013 as shown in Fig. 6.6.

On June 4, 2013, two "**BBC News**" captioned "**Gay marriage bill: Lord debate wrecking amendment**" and "**QA: Gay marriage**" were mapped to *#gayrights* ⇒ *#gaymarriage*. The former was reported at 2:10am and on the same day at 17:03 another update captioned "**Gay marriage paves was for polygamy, says Lord Carey**" was mapped to this evolving rule. Within 24 hours, another update captioned "**Gay marriage bill: Peers back government plans**" was also mapped to our system detection. On June 27, 2013 at 02:53 an update captioned "**US Supreme Court in historic rulings on gay marriage**" reported by "**BBC US and Canada news**" was mapped to *#politicalparties* ⇒ *#gaymarriage*. Later on the same day "**The BBC, Scotland**" updated a related news captioned "**Scotland's gay marriage bill published at Holyrood**" as presented in Fig. 6.6. The time of release of all the news updates in real-life matched with the *time windows* of the rule evolvements detected by RTI-*Mapping*.

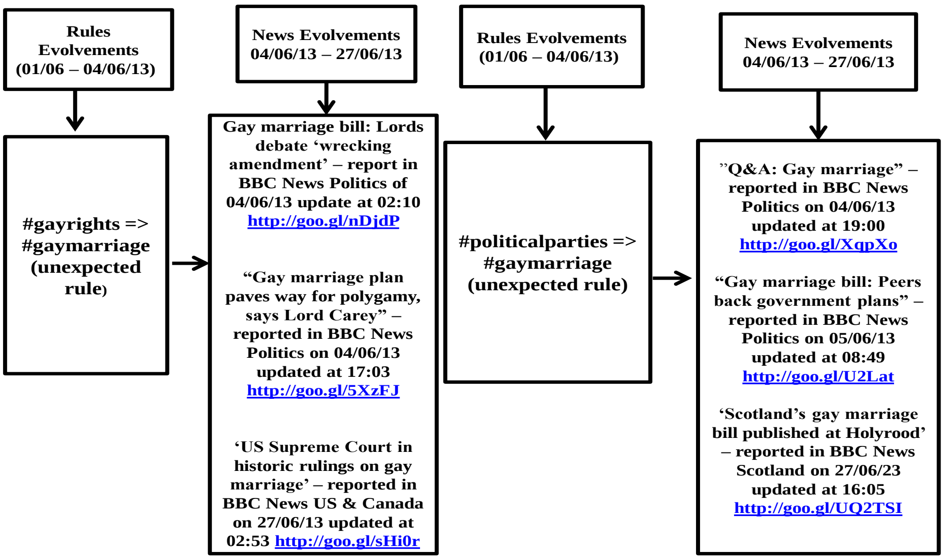

Rules Evolvements (01/06 – 04/06/13)
#gayrights => #gaymarriage (unexpected rule)
News Evolvements 04/06/13 – 27/06/13
Gay marriage bill: Lords debate 'wrecking amendment' – report in BBC News Politics of 04/06/13 update at 02:10 http://goo.gl/nDjdP
"Gay marriage plan paves way for polygamy, says Lord Carey" – reported in BBC News Politics on 04/06/13 updated at 17:03 http://goo.gl/5XzFJ
'US Supreme Court in historic rulings on gay marriage' – reported in BBC News US & Canada on 27/06/13 updated at 02:53 http://goo.gl/sHi0r
Rules Evolvements (01/06 – 04/06/13)
#politicalparties => #gaymarriage (unexpected rule)
News Evolvements 04/06/13 – 27/06/13
"Q&A: Gay marriage" – reported in BBC News Politics on 04/06/13 updated at 19:00 http://goo.gl/XqpXo
"Gay marriage bill: Peers back government plans" – reported in BBC News Politics on 05/06/13 updated at 08:49 http://goo.gl/U2Lat
'Scotland's gay marriage bill published at Holyrood' – reported in BBC News Scotland on 27/06/23 updated at 16:05 http://goo.gl/UQ2TSI

FIGURE 6.6: Rule Mapping for *#GayMarriage*

6.6.4 Case Study 4 - *#Shutdown*

From October 1 to October 16, 2013, the United States federal government went into **shutdown** to restrict most routine operations. This was because the Congress failed to enact legislation appropriating funds for fiscal year 2014, or a continuing resolution for the interim authorisation of appropriations for fiscal year 2014. We conducted an RTI-*Mapping* experiments to analyse the event as reported on Twitter network and to map *ARs* identified by our system to news headlines of newsagents in real-life. We specify #Shutdown to crawl Twitter using its *API*. We extracted related hashtaged tweets posted on-line from October 1 to October 4, 2013. A number of *emerging* and *unexpected* rules were detected by our system. As expected, all the rules discovered pointed to the US government shutdown. *#government* ⇒ *#shutdown* was mapped to different *CNN* news reports and updates as presented in Fig. 6.7.

On October 1, 2013, two news headlines captioned **"Shutdown: What happen next?"** and "**U.S. Government Shuts Down as Congress can't Agree on Spending Bill**" were mapped to *#government* ⇒ *#shutdown* . The first was reported on *CNNMoney* less than an hour into the US shutdown while the second was reported on *CNN politics* about 3 hours later. On November 8, 2013 another update on *CNN politics* was mapped to *#government* ⇒ *#shutdown*. *AR #Shutdown* ⇒ *#PlannedParenthood* evolved unexpectedly and was mapped to *CNN* news captioned "**In Shutdowns, an Attack on Women's Health**" that was reported on October 1, 2013 at 1402 GMT.

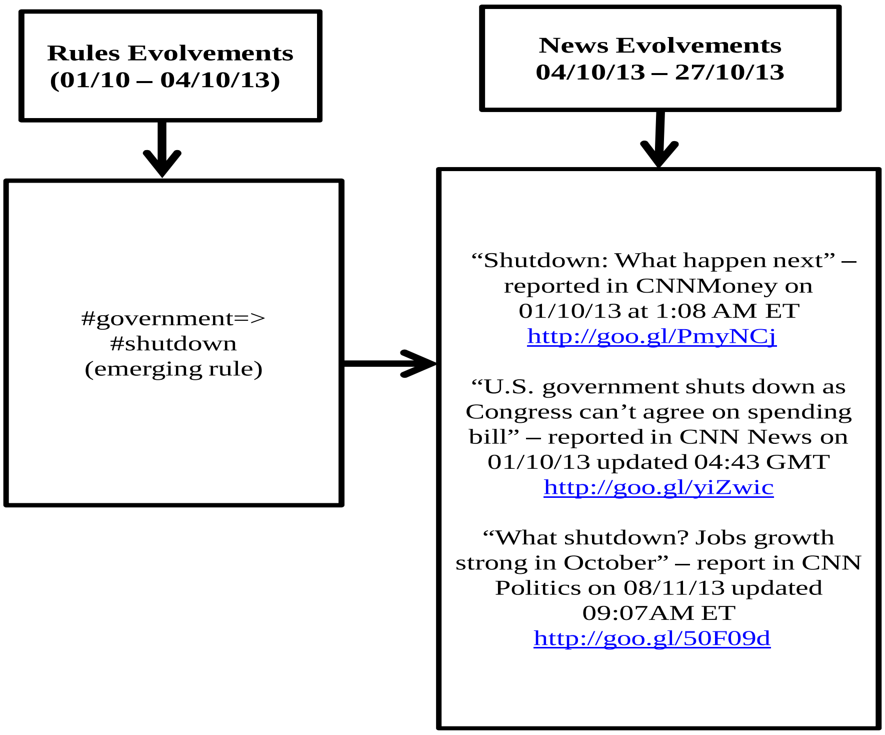

Rules Evolvements
(01/10 – 04/10/13)
News Evolvements
04/10/13 – 27/10/13
#government=>
#shutdown
(emerging rule)
"Shutdown: What happen next" – reported in CNNMoney on 01/10/13 at 1:08 AM ET
http://goo.gl/PmyNCj
"U.S. government shuts down as Congress can't agree on spending bill" – reported in CNN News on 01/10/13 updated 04:43 GMT
http://goo.gl/yiZwic
"What shutdown? Jobs growth strong in October" – report in CNN Politics on 08/11/13 updated 09:07AM ET
http://goo.gl/50F09d

FIGURE 6.7: Rule Mapping for *#Shutdown*

6.6.5 Case Study 5 - *#BusinessNews*

Business news, unlike the other four case studies is not based on any event. We chose this domain to analyse how RTI-*Mapping* will handle *ARs* in tweet hashtags of a topic that is constantly visible as part of other news posted on Twitter. Business news is reported by traditional newsagents all over the world and most business related tweets are often labelled with *#BusinessNews*. The *ARs* detected in RTI-*Mapping* experiments conducted on *#BusinessNews* are mapped to business news on *BBC News*. As shown in Fig. 6.8, some of the news that are related to *#BusinessNews* were on going before the experiments commenced. However, *#SprintNextel* ⇒ *#BusinessNews* evolved unexpectedly on Twitter during the experiments, this was attributed to the take-over of Sprint corporation (the biggest intercom corporation in the US) by a Japanese company (Softbank) at the time.

On May 29, 2013, *BBC news* reported on *Softbank* getting the US approval for national security clearance to buy 70% stake in Sprint Nextel for $20.1bn under the caption "**Sprint-Softbank gets US national security clearance**". An update to the story was mapped to *#SprintNextel* ⇒ *#BusinessNews* on June 11, 2013 with the caption, "**Softbank Sweetens Offer for Sprint Nextel Shareholders**". On June 19, 2013 the story evolved resulting in the news update captioned "**Dish Network abandons bid for Sprint Nextel**". Consequently on June 26, 2013 another update was spotted with caption "**Sprint Nextel shareholders approve Softbank Bid**". The last mapping in the case study was completed on October 12, 2013 with headline captioned "**Softbank Shares Plunge on News of Sprint Nextel talks**". All the news items mapped include keywords **Sprint Nextel** and **business news** detected by our system as evolving *ARs* in related tweets within the specific Time Frame Windows (TFWs).

In Section 6.7 we visualise hashtags and their corresponding rules in *#Boston* dataset using **TRCM-viz**.

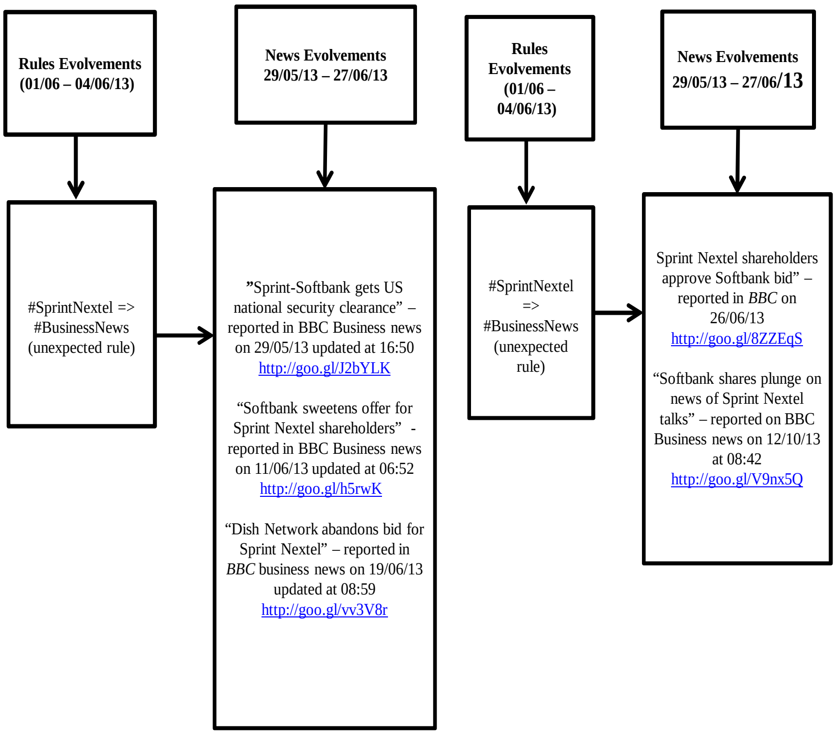

Rules Evolvements (01/06 – 04/06/13)
#SprintNextel => #BusinessNews (unexpected rule)
News Evolvements 29/05/13 – 27/06/13
"Sprint-Softbank gets US national security clearance" – reported in BBC Business news on 29/05/13 updated at 16:50 http://goo.gl/J2bYLK
"Softbank sweetens offer for Sprint Nextel shareholders" - reported in BBC Business news on 11/06/13 updated at 06:52 http://goo.gl/h5rwK
"Dish Network abandons bid for Sprint Nextel" – reported in BBC business news on 19/06/13 updated at 08:59 http://goo.gl/vv3V8r
Rules Evolvements (01/06 – 04/06/13)
#SprintNextel => #BusinessNews (unexpected rule)
News Evolvements 29/05/13 – 27/06/13
Sprint Nextel shareholders approve Softbank bid" – reported in BBC on 26/06/13 http://goo.gl/8ZZEqS
"Softbank shares plunge on news of Sprint Nextel talks" – reported on BBC Business news on 12/10/13 at 08:42 http://goo.gl/V9nx5Q

FIGURE 6.8: Rule Mapping for #*Businessnews*

6.7 Filtering Interesting Hashtags Using TRCM-Viz

Human analysis of *ARs* from a large database often leave analysts with the burden of scanning through large number of rules to extract useful ones [Hahsler and Chelluboina, 2011]. A graph-based NoSQL visualisation can help lighten the burden. Visualisation can be used to trim down rules discovered in large database such that only specific rules are visible at a time. This is a method of zooming-in on crowded data to pinpoint specific rules for analysis. It is commonly used to communicate intangible and tangible ideas in different field such as science, engineering and education [Prangsma et al., 2009]. It can also be used to analyse tangible real-life tasks [Balakrishnan and Ranganathan, 2012]. In this chapter we used graph theory for visualisation of ARs. A graph is made up of *nodes* linked with *edges* which connect some of the nodes [Chen, 2012, Deo, 2004]. Each edge is linked with other disconnected pair of nodes. Points denote the nodes while the edges are links to the nodes (as shown in Fig. 6.9). A node that has connection with an edge are known as endpoints of the edge. A node in a graph may be disjointed from the edges in the same graph.

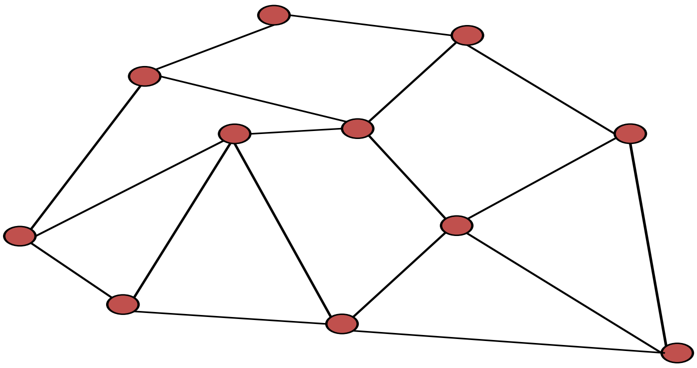
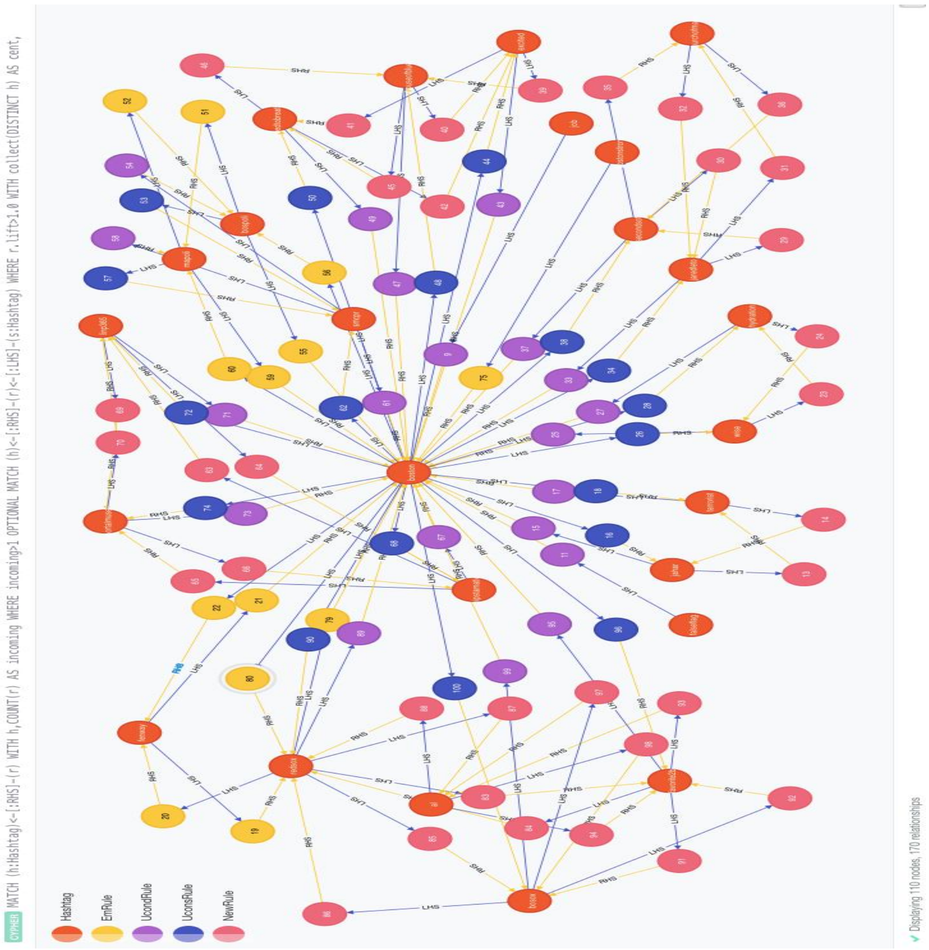


Figure 6.9: Nodes and Edges in Simple Graph

Different AR visualisation methods have been applied to large databases to filter useful rules [Fukuda et al., 1996,Hahsler and Chelluboina, 2011,Liu et al., 2012,Wong et al., 1999,Ong et al., 2002]. The work of [Hahsler and Chelluboina, 2011] proposed the Matrix-based visualisation method, where the antecedent and consequent itemsets are arranged on the x and y axes. They defined interest measure, which is displayed at the intersection of the antecedent and consequent of rules under consideration. The intersection is left blank if there is no rule in the antecedent/consequent.

In this section we propose our novel visualisation method named **TRCM-Viz** using NoSQL database system named **Neo4j**. NoSQL is becoming popular in the sphere of developers of Web 2.0 applications since they perform better than traditional relational databases [Holzschuher and Peinl, 2013]. *Neo4j* is an open-source graph database which is developed in *Java* [6]. It has proved to be a prominent graph database which is faster than relational databases in many applications [Webber, 2012]. We adopted **Neo4j** to build visualisation of aforementioned #*Boston* dataset. Our aim is to build different clusters of nodes instead of going through a collection of clusters which makes little or no meaning due to several nodes and links interlocking one another as presented in Fig 6.10 .



```
MATCH (h:Hashtag)<-[:RHS]-(r) WITH h,COUNT(r) AS incoming WHERE incoming>1 OPTIONAL MATCH (h)<-[:RHS]-(r)<-[:LHS]-(s:Hashtag) WHERE r.lift>1.0 WITH collect(DISTINCT h) AS cent,
```

FIGURE 6.10: Visualising all the Association Rules Present in #*Boston*

[6] http://neo4j.org

We defined a **cluster** in a graph using a **centroid-based** notation, the number of links to a specific node (hashtag) can determine whether this hashtag is a centroid. Any hashtag node that exceeds a given threshold value for the number of links is considered a centroid. Together with all the connections of this particular hashtag, a cluster is formed. Only strong connections are formed using a rule interestingness measure known as **lift**. We discussed lift measure in Chapter 3. However, to improve readability, lift can be calculated using the following formula:

$$lift(A \Rightarrow B) = \frac{confidence(A \Rightarrow B)}{confidence(\emptyset \Rightarrow B)} = \frac{support(A \Rightarrow B)}{support(A) \times support(B)} \tag{6.1}$$

In this visualisation we are interested in three important axioms: 1) the nodes and edges that forms a centroid and subsequently the centroid that forms a clusters; 2) the threshold for number of links to define a centroid (this is mentioned later in the section); and 3) not all links are considered because we pruned weaker rules with the lift. *Neo4j* is used to present rules in the #*Boston* dataset as nodes and links clusters, making visualisation easy to analyse. The nodes are represented by “hashtags” and “TRCM” rules. The relationships between hashtag and rule nodes are “LHS” or “RHS”. Based on this graph, the centroids of the clusters of the dataset focused on the hashtags with largest number of incoming links from rules as presented in table 6.2. The centroid is the main/-most connected hashtag(s) in each cluster. As presented in Fig 6.11, the main centroids in the two clusters are made up of #*Boston* and #*Redsox*. The two clusters are shown separately in Fig 6.12 and Fig 6.13 for clearer visualisation.

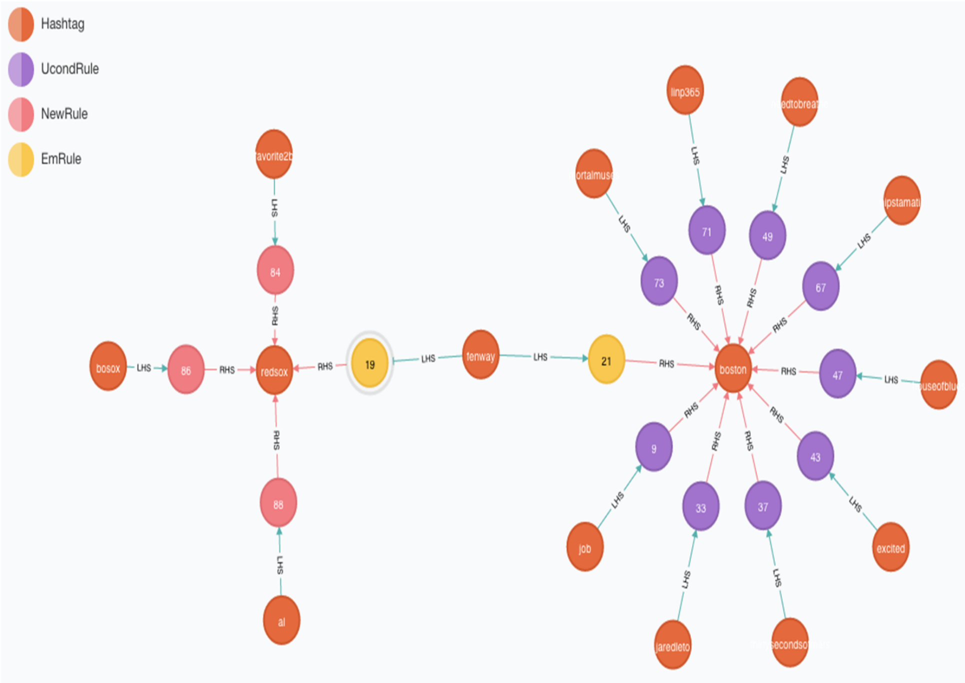


FIGURE 6.11: Visualising #*Boston* in Clusters

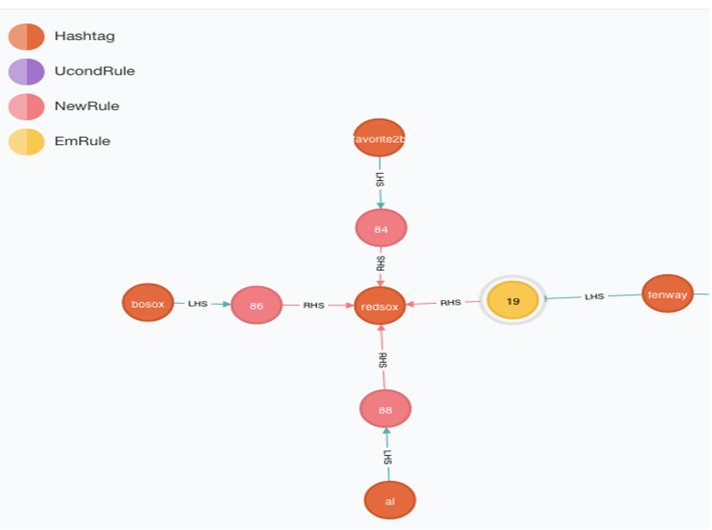


FIGURE 6.12: Visualising Part 1 of #*Boston* Clusters

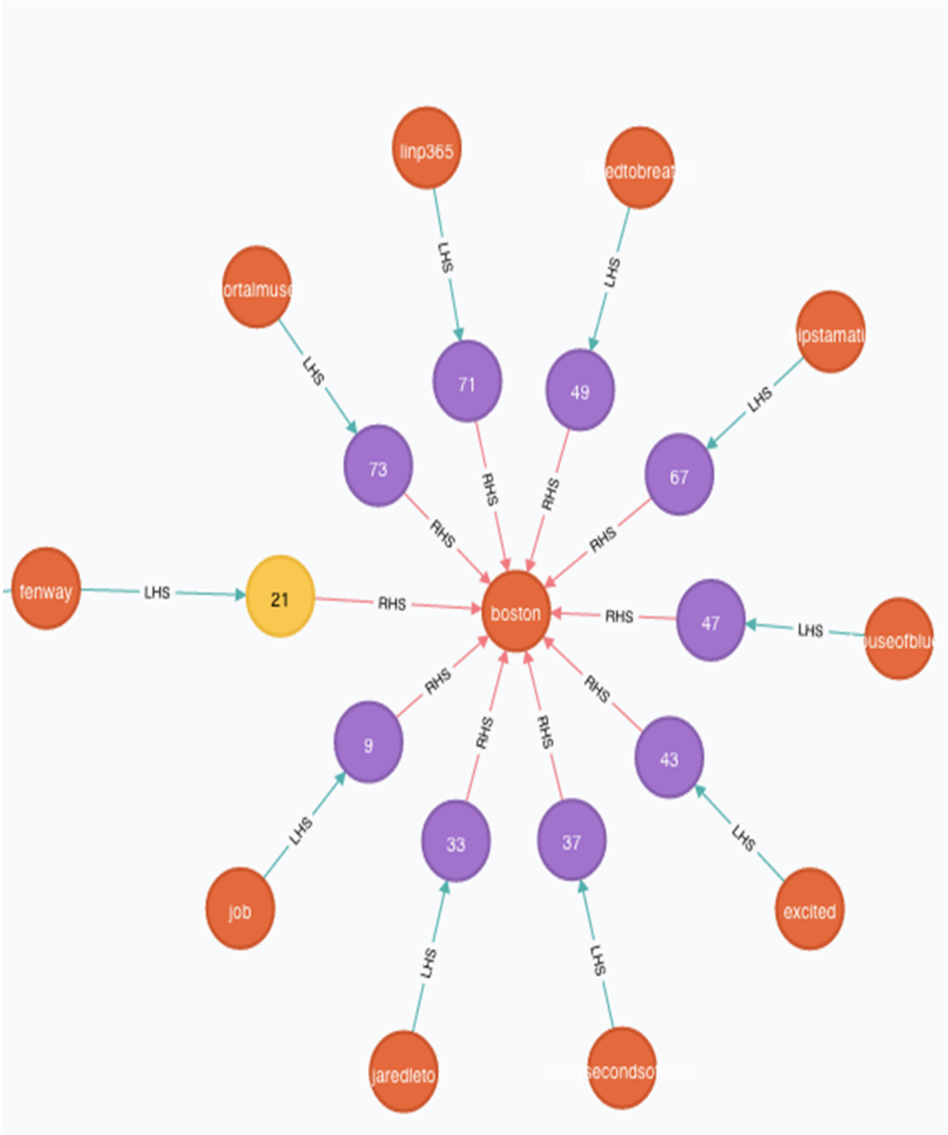


FIGURE 6.13: Visualising Part 2 of #*Boston* Clusters

For the event tracking experiment, we visualised *rules* and their corresponding *hashtags* in the #*Boston* dataset as presented in Fig 6.11 and 6.15 which consist of 29 and 10 nodes respectively. As shown in table 6.2, the first two columns correspond to the threshold values for the incoming nodes and lift when they are being incremented. In Fig 6.11 and 6.15, the threshold of incoming nodes was

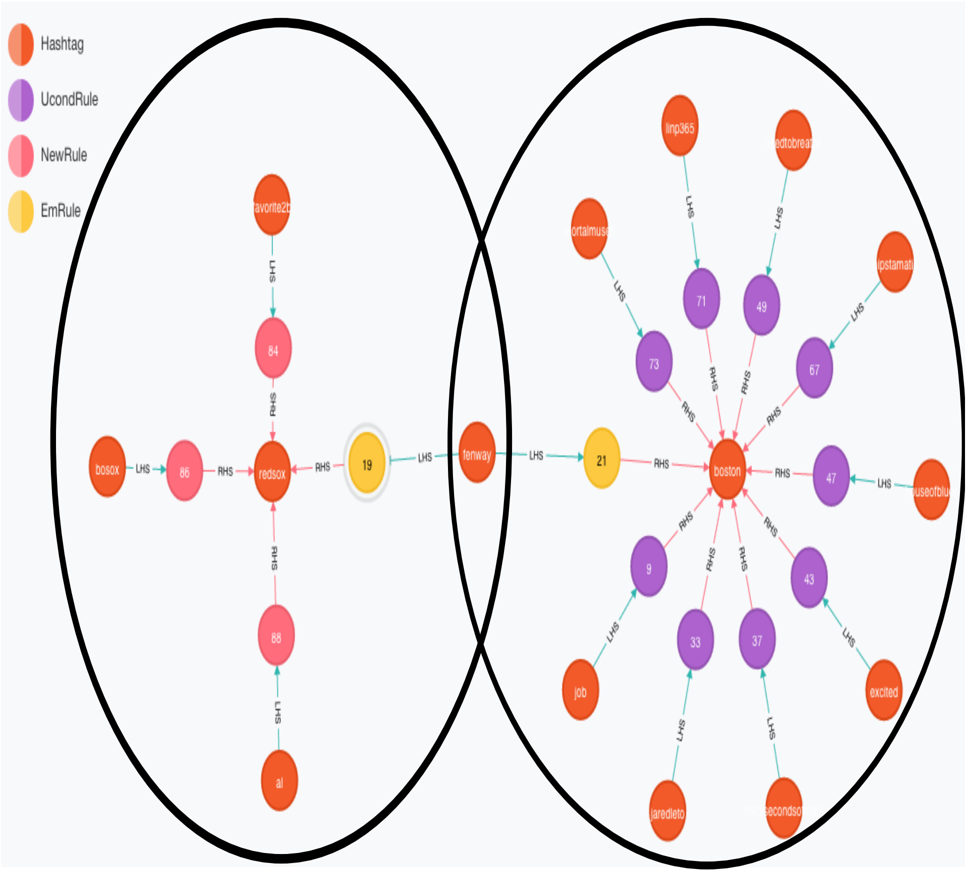


Figure 6.14: *#Fenway* Sharing Two Clusters

incremented to 4, while the threshold for the lift was incremented to 1.92. The total number of nodes was subsequently reduced to 29 which makes it easier to visualise the cluster. The third column is the number of centroid nodes. The fourth column shows the number of relevant hashtag nodes associated to the previous centroid nodes. In the case of *#Fenaway* its length as shown in Table 6.2 is 19 which corresponds to its location in Fig 6.17 and can be viewed in the *lhs* of *#Redsox*. The rule evolves as *emerging* rule represented with colour code yellow. The fifth column represents the number of rule nodes. *#needtobreath* has a rule node of 49 and can be viewed in the *lhs* of *#Boston* evolving as *unexpected* rule represented with colour code purple). The last column is the summation of the previous three columns, which represents the total number of nodes in the clusters. The visualisation of the $\#Boston$ dataset comprises of two main clusters. Fig 6.14 show $\#Fenaway$ sharing the two clusters, it can be viewed in the *lhs* of $\#Boston$ and $\#Redsox$. This demonstrates the interestingness of $\#Fenaway$ in the two clusters. Fig. 6.15 shows that $\#Redsox$ has centroid length of 4 namely: $\#Fenaway$, $\#al$, $\#favorite2B$ and $\#Bosox$, all the rules except $\#Fenaway$ can be viewed as *new* rules.

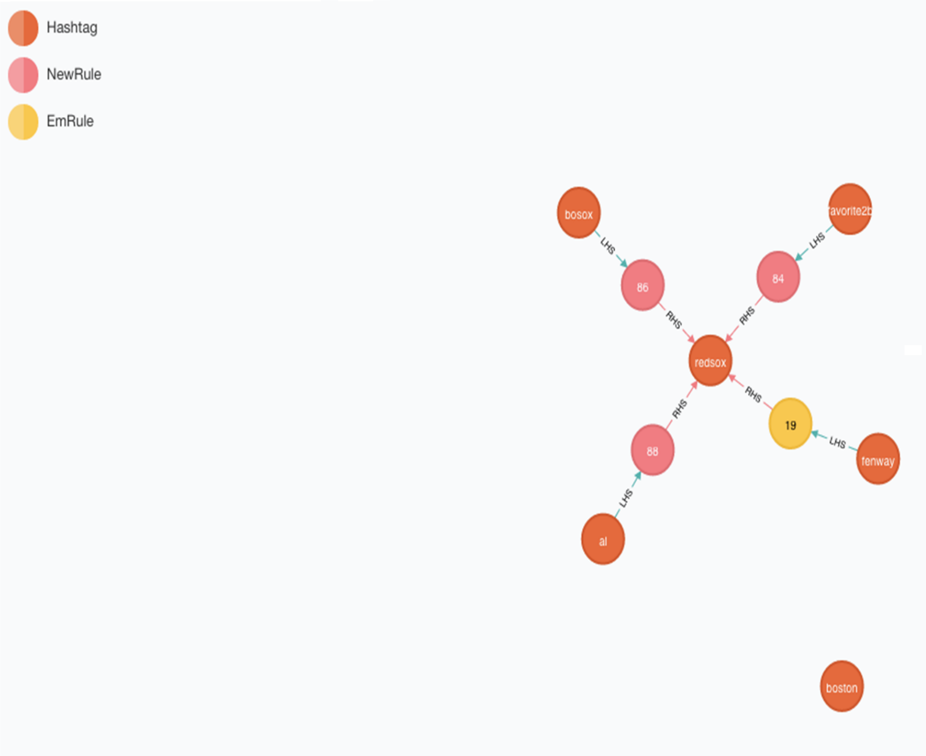
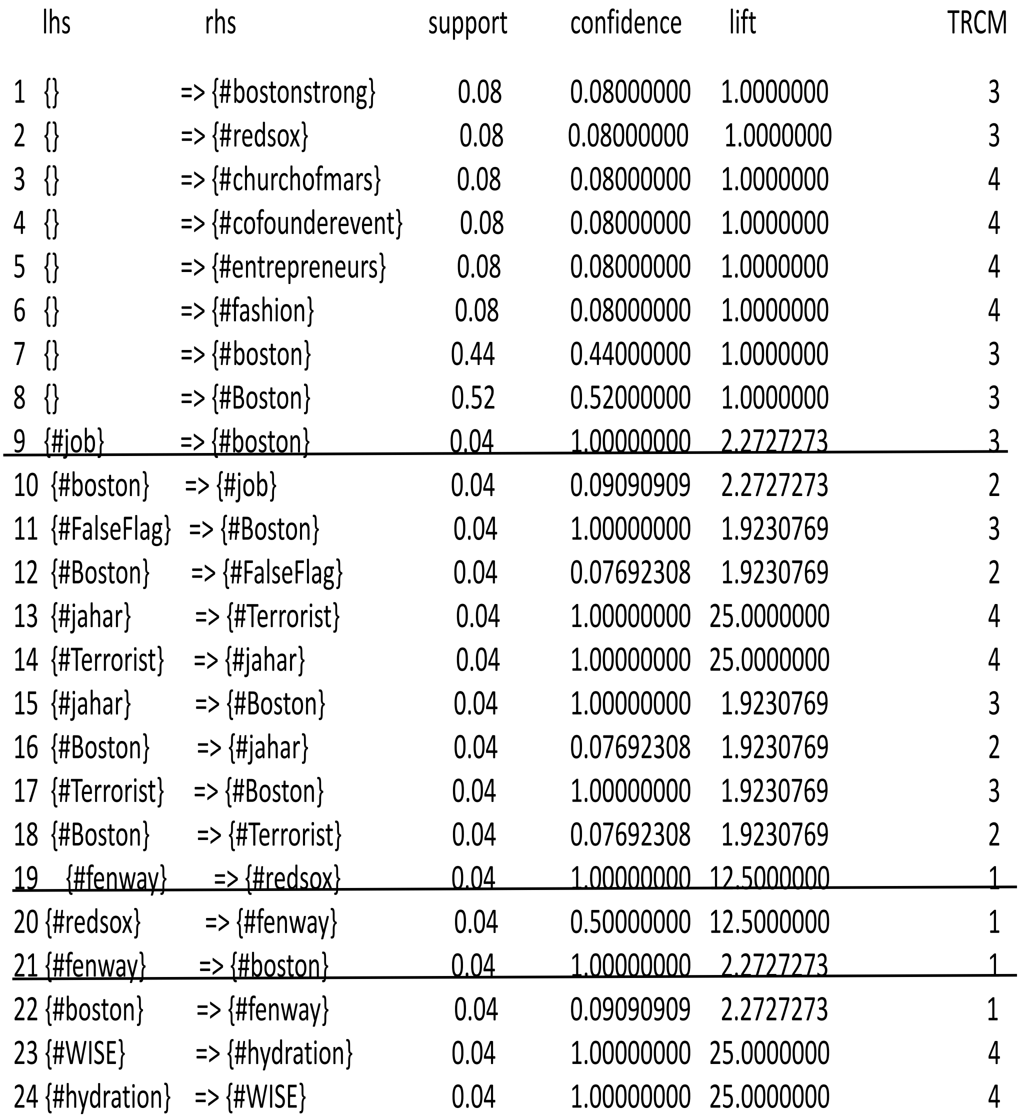


Figure 6.15: Visualising the Interestingness of $\#Redsox$

Visualising the detected rules in clusters as presented in Figs. 6.11, 6.14 and 6.15 allow analysts the convenience of viewing each rules in clusters and discovering the interestingness of hashtags present in each centroid as well as the strength of such centroid.

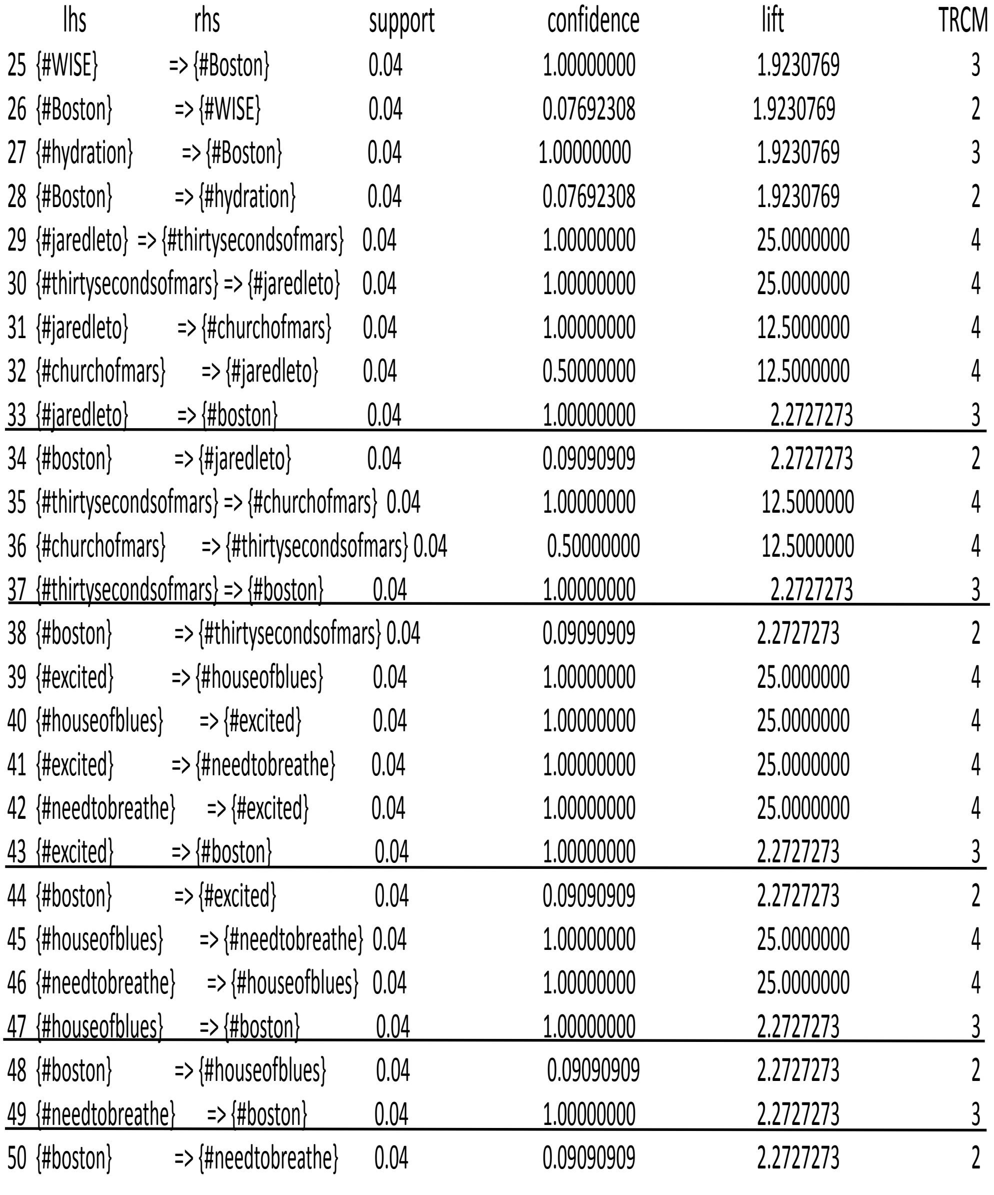
Table 6.2: Boston Dataset Visualisation

Incoming	Lift	Centroid Length	Relevant Hashtags Length	Rules Length	Total Nodes
1	1.0	22	3	85	110
2	1.0	16	9	73	98
2	1.1363636	16	8	70	94
3	1.1363636	4	19	34	57
3	1.9230769	4	11	20	35
4	1.9230769	2	13	14	29
4	2.2727273	2	4	4	10



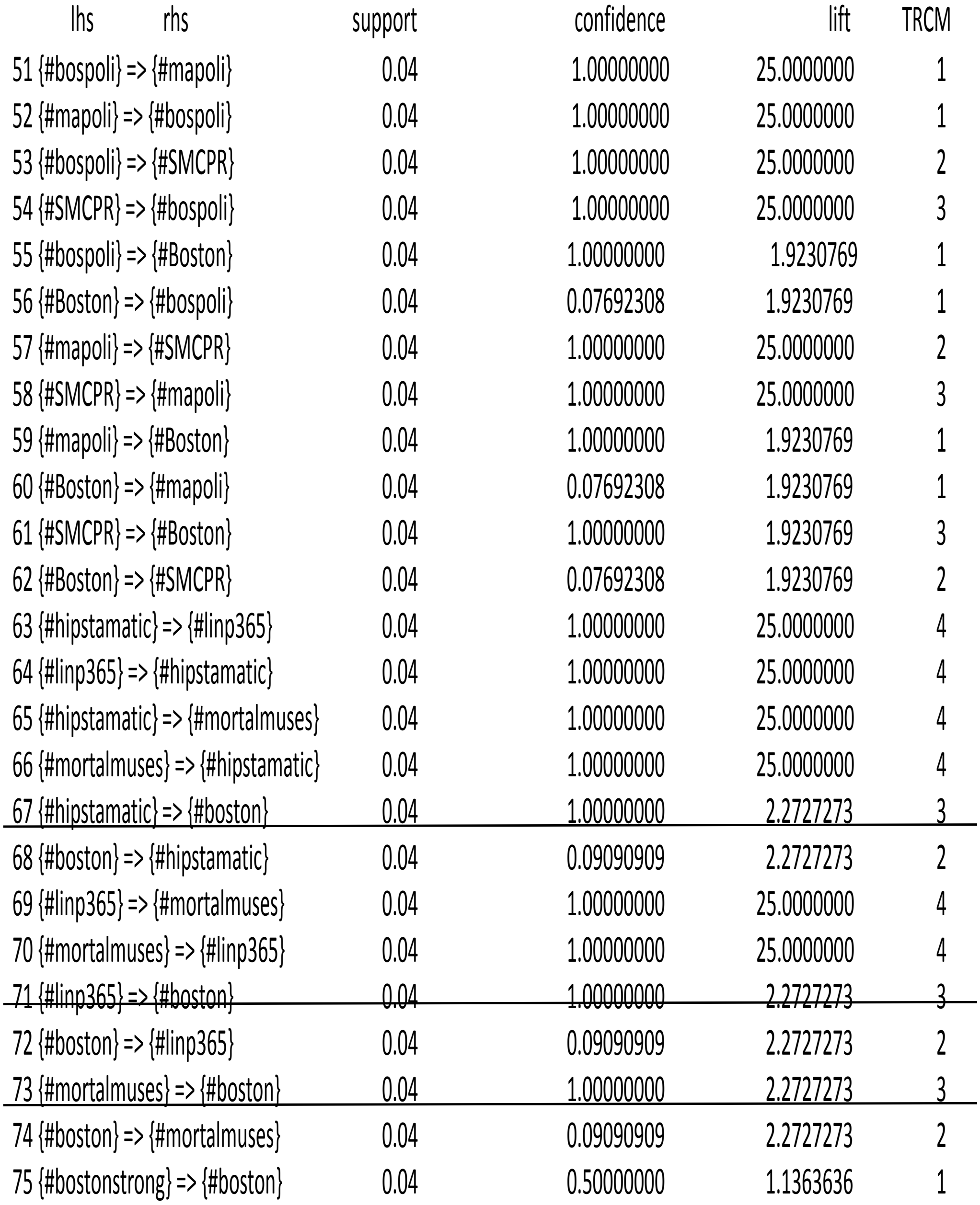
	lhs	rhs	support	confidence	lift	TRCM
1	{}	=> {#bostonstrong}	0.08	0.08000000	1.0000000	3
2	{}	=> {#redsox}	0.08	0.08000000	1.0000000	3
3	{}	=> {#churchofmars}	0.08	0.08000000	1.0000000	4
4	{}	=> {#cofounderevent}	0.08	0.08000000	1.0000000	4
5	{}	=> {#entrepreneurs}	0.08	0.08000000	1.0000000	4
6	{}	=> {#fashion}	0.08	0.08000000	1.0000000	4
7	{}	=> {#boston}	0.44	0.44000000	1.0000000	3
8	{}	=> {#Boston}	0.52	0.52000000	1.0000000	3
9	{#job}	=> {#boston}	0.04	1.00000000	2.2727273	3
10	{#boston}	=> {#job}	0.04	0.09090909	2.2727273	2
11	{#FalseFlag}	=> {#Boston}	0.04	1.00000000	1.9230769	3
12	{#Boston}	=> {#FalseFlag}	0.04	0.07692308	1.9230769	2
13	{#jahar}	=> {#Terrorist}	0.04	1.00000000	25.0000000	4
14	{#Terrorist}	=> {#jahar}	0.04	1.00000000	25.0000000	4
15	{#jahar}	=> {#Boston}	0.04	1.00000000	1.9230769	3
16	{#Boston}	=> {#jahar}	0.04	0.07692308	1.9230769	2
17	{#Terrorist}	=> {#Boston}	0.04	1.00000000	1.9230769	3
18	{#Boston}	=> {#Terrorist}	0.04	0.07692308	1.9230769	2
19	{#fenway}	=> {#redsox}	0.04	1.00000000	12.5000000	1
20	{#redsox}	=> {#fenway}	0.04	0.50000000	12.5000000	1
21	{#fenway}	=> {#boston}	0.04	1.00000000	2.2727273	1
22	{#boston}	=> {#fenway}	0.04	0.09090909	2.2727273	1
23	{#WISE}	=> {#hydration}	0.04	1.00000000	25.0000000	4
24	{#hydration}	=> {#WISE}	0.04	1.00000000	25.0000000	4

FIGURE 6.16: Figure Showing the Association Rules in #*Boston*



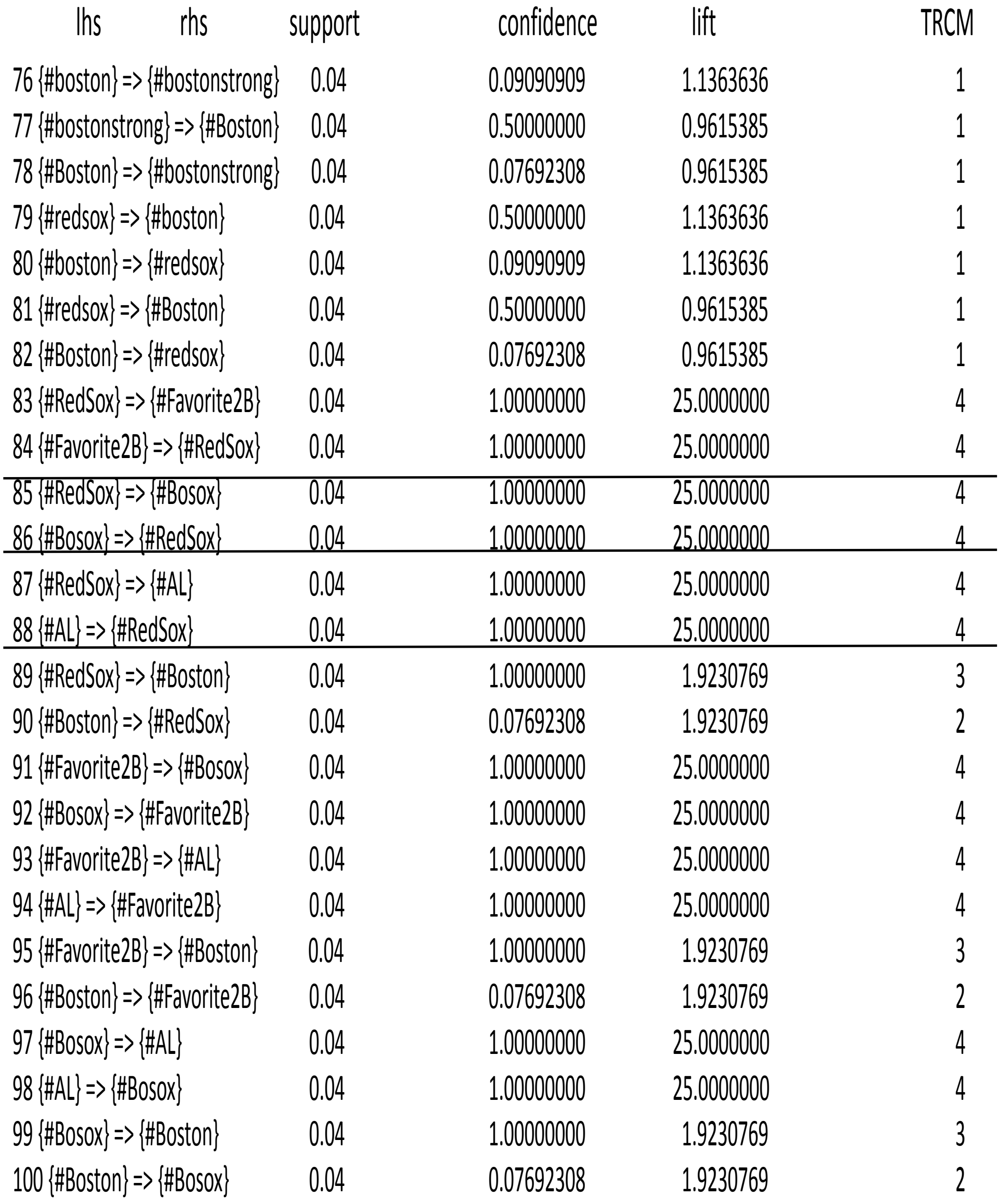
	lhs		rhs	support	confidence	lift	TRCM
25	{#WISE}	=>	{#Boston}	0.04	1.00000000	1.9230769	3
26	{#Boston}	=>	{#WISE}	0.04	0.07692308	1.9230769	2
27	{#hydration}	=>	{#Boston}	0.04	1.00000000	1.9230769	3
28	{#Boston}	=>	{#hydration}	0.04	0.07692308	1.9230769	2
29	{#jaredleto}	=>	{#thirtysecondsofmars}	0.04	1.00000000	25.0000000	4
30	{#thirtysecondsofmars}	=>	{#jaredleto}	0.04	1.00000000	25.0000000	4
31	{#jaredleto}	=>	{#churchofmars}	0.04	1.00000000	12.5000000	4
32	{#churchofmars}	=>	{#jaredleto}	0.04	0.50000000	12.5000000	4
33	{#jaredleto}	=>	{#boston}	0.04	1.00000000	2.2727273	3
34	{#boston}	=>	{#jaredleto}	0.04	0.09090909	2.2727273	2
35	{#thirtysecondsofmars}	=>	{#churchofmars}	0.04	1.00000000	12.5000000	4
36	{#churchofmars}	=>	{#thirtysecondsofmars}	0.04	0.50000000	12.5000000	4
37	{#thirtysecondsofmars}	=>	{#boston}	0.04	1.00000000	2.2727273	3
38	{#boston}	=>	{#thirtysecondsofmars}	0.04	0.09090909	2.2727273	2
39	{#excited}	=>	{#houseofblues}	0.04	1.00000000	25.0000000	4
40	{#houseofblues}	=>	{#excited}	0.04	1.00000000	25.0000000	4
41	{#excited}	=>	{#needtobreathe}	0.04	1.00000000	25.0000000	4
42	{#needtobreathe}	=>	{#excited}	0.04	1.00000000	25.0000000	4
43	{#excited}	=>	{#boston}	0.04	1.00000000	2.2727273	3
44	{#boston}	=>	{#excited}	0.04	0.09090909	2.2727273	2
45	{#houseofblues}	=>	{#needtobreathe}	0.04	1.00000000	25.0000000	4
46	{#needtobreathe}	=>	{#houseofblues}	0.04	1.00000000	25.0000000	4
47	{#houseofblues}	=>	{#boston}	0.04	1.00000000	2.2727273	3
48	{#boston}	=>	{#houseofblues}	0.04	0.09090909	2.2727273	2
49	{#needtobreathe}	=>	{#boston}	0.04	1.00000000	2.2727273	3
50	{#boston}	=>	{#needtobreathe}	0.04	0.09090909	2.2727273	2

FIGURE 6.17: Figure Showing the Association Rules in #*Boston*



	lhs		rhs	support	confidence	lift	TRCM
51	{#bospoli}	=>	{#mapoli}	0.04	1.00000000	25.0000000	1
52	{#mapoli}	=>	{#bospoli}	0.04	1.00000000	25.0000000	1
53	{#bospoli}	=>	{#SMCPR}	0.04	1.00000000	25.0000000	2
54	{#SMCPR}	=>	{#bospoli}	0.04	1.00000000	25.0000000	3
55	{#bospoli}	=>	{#Boston}	0.04	1.00000000	1.9230769	1
56	{#Boston}	=>	{#bospoli}	0.04	0.07692308	1.9230769	1
57	{#mapoli}	=>	{#SMCPR}	0.04	1.00000000	25.0000000	2
58	{#SMCPR}	=>	{#mapoli}	0.04	1.00000000	25.0000000	3
59	{#mapoli}	=>	{#Boston}	0.04	1.00000000	1.9230769	1
60	{#Boston}	=>	{#mapoli}	0.04	0.07692308	1.9230769	1
61	{#SMCPR}	=>	{#Boston}	0.04	1.00000000	1.9230769	3
62	{#Boston}	=>	{#SMCPR}	0.04	0.07692308	1.9230769	2
63	{#hipstamatic}	=>	{#linp365}	0.04	1.00000000	25.0000000	4
64	{#linp365}	=>	{#hipstamatic}	0.04	1.00000000	25.0000000	4
65	{#hipstamatic}	=>	{#mortalmuses}	0.04	1.00000000	25.0000000	4
66	{#mortalmuses}	=>	{#hipstamatic}	0.04	1.00000000	25.0000000	4
67	{#hipstamatic}	=>	{#boston}	0.04	1.00000000	2.2727273	3
68	{#boston}	=>	{#hipstamatic}	0.04	0.09090909	2.2727273	2
69	{#linp365}	=>	{#mortalmuses}	0.04	1.00000000	25.0000000	4
70	{#mortalmuses}	=>	{#linp365}	0.04	1.00000000	25.0000000	4
~~71~~	~~{#linp365}~~	~~=>~~	~~{#boston}~~	~~0.04~~	~~1.00000000~~	~~2.2727273~~	~~3~~
72	{#boston}	=>	{#linp365}	0.04	0.09090909	2.2727273	2
73	{#mortalmuses}	=>	{#boston}	0.04	1.00000000	2.2727273	3
74	{#boston}	=>	{#mortalmuses}	0.04	0.09090909	2.2727273	2
75	{#bostonstrong}	=>	{#boston}	0.04	0.50000000	1.1363636	1

Figure 6.18: Figure Showing the Association Rules in *#Boston*



	lhs		rhs	support	confidence	lift	TRCM
76	{#boston}	=>	{#bostonstrong}	0.04	0.09090909	1.1363636	1
77	{#bostonstrong}	=>	{#Boston}	0.04	0.50000000	0.9615385	1
78	{#Boston}	=>	{#bostonstrong}	0.04	0.07692308	0.9615385	1
79	{#redsox}	=>	{#boston}	0.04	0.50000000	1.1363636	1
80	{#boston}	=>	{#redsox}	0.04	0.09090909	1.1363636	1
81	{#redsox}	=>	{#Boston}	0.04	0.50000000	0.9615385	1
82	{#Boston}	=>	{#redsox}	0.04	0.07692308	0.9615385	1
83	{#RedSox}	=>	{#Favorite2B}	0.04	1.00000000	25.0000000	4
84	{#Favorite2B}	=>	{#RedSox}	0.04	1.00000000	25.0000000	4
85	{#RedSox}	=>	{#Bosox}	0.04	1.00000000	25.0000000	4
86	{#Bosox}	=>	{#RedSox}	0.04	1.00000000	25.0000000	4
87	{#RedSox}	=>	{#AL}	0.04	1.00000000	25.0000000	4
88	{#AL}	=>	{#RedSox}	0.04	1.00000000	25.0000000	4
89	{#RedSox}	=>	{#Boston}	0.04	1.00000000	1.9230769	3
90	{#Boston}	=>	{#RedSox}	0.04	0.07692308	1.9230769	2
91	{#Favorite2B}	=>	{#Bosox}	0.04	1.00000000	25.0000000	4
92	{#Bosox}	=>	{#Favorite2B}	0.04	1.00000000	25.0000000	4
93	{#Favorite2B}	=>	{#AL}	0.04	1.00000000	25.0000000	4
94	{#AL}	=>	{#Favorite2B}	0.04	1.00000000	25.0000000	4
95	{#Favorite2B}	=>	{#Boston}	0.04	1.00000000	1.9230769	3
96	{#Boston}	=>	{#Favorite2B}	0.04	0.07692308	1.9230769	2
97	{#Bosox}	=>	{#AL}	0.04	1.00000000	25.0000000	4
98	{#AL}	=>	{#Bosox}	0.04	1.00000000	25.0000000	4
99	{#Bosox}	=>	{#Boston}	0.04	1.00000000	1.9230769	3
100	{#Boston}	=>	{#Bosox}	0.04	0.07692308	1.9230769	2

Figure 6.19: Figure Showing the Association Rules in #*Boston*

Having discussed the visualisation of rules that showed qualitatively interesting hashtags, the next section will provide a brief qualitative performance analysis of all the case studies and the visualization of rules in the #*Boston* dataset.

6.8 Qualitative Performance Analysis

RTI-*Mapping* portrays the significance of rule evolvements in tweets and how rule evolvements can be tracked in unfolding news reports and news updates in real-life. All the case studies adopted in the experiments conducted in this chapter were widely reported both on Twitter and on traditional news media. The experiments have been able to show that *RTI* is capable of mapping real-life news within specific time windows by way of comparing identified *ARs* in our experiments to the news of traditional newsagents such as **The Telegraph, BBC and CNN**. RTI-*Mapping* was also able to track the news updates as they unfold in the news. These experiments underscore the relevance of tweets' hashtags on Twitter. *RTI* can be used to filter hashtags to attract users' attention to current news. Rule evolvements can also be used to update hashtags used in tweets at different evolvement phases. Lastly, we have shown through *TRCM-viz*, a better visualisation of *TRCM* rules that help analyts make better use of *ARs*.

6.9 Summary

This chapter adopted a novel methodology termed RTI-*Mapping* to map and track *ARs* identified in tweets' hashtags at a specific *TFWs* to evolving news reported by on-line traditional newsagents. RTI-*Mapping* can be used to enhance news updates if hashtags are updated timely and appropriately. Mapping *ARs* evolvements to news evolvements in reality are one of the factors of measuring the reliability of Twitter data.

We presented a qualitative study of *TRCM* for event tracking in real-life and applied *TRCM-RTI* on tweets of widely reported topics/events from different domains. The domains covered include social, politics, socioeconomic and business during a specific time windows. We mapped and tracked the evolvements of hashtag keywords identified by *TRCM-RTI* to the evolvements of related news and news updates in real-life. We showed how the concept of "TwO - NwO" state could be applied to RTI-*Mapping*.

Finally, we presented the visualisation of #*Boston* dataset using our novel **TRCM-Viz** built with NoSQL database system named *Neo4j* to demonstrate the relationships between hashtags and rules (nodes). The visualisation also showed "(centroids)" and their related links.

The application of our method quantitatively (in Chapter 5) and qualitatively (in Chapter 6 to real case studies buttress the efficiency of our method as a *TDT* tool for Twitter data. The case studies reported in this chapter demonstrated how *TRCM-RTI* can be used to analyse chain in topic/event evolvements in reality. The benefits of our method include the enhancement of trending topics on Twitter. RTI-*Mapping* also enables rule trace and assists Twitter users to understand the origin of rules and their evolvement chain.

Chapter 7 concludes the research reported in this thesis by highlighting the work carried out in each of the chapters. The chapter also states possible future work of the research.

Chapter 7

Conclusion and Future Work

This chapter gives the general conclusion and highlights of each chapter in this thesis. The work carried out in this thesis presented a comprehensive and original research on the application of a novel methodology termed **Transaction-based Rule Change Mining (TRCM)** on Twitter data for Topic Detection and Tracking *(TDT)*. Hashtag is most commonly used in tweets when compared to its usage on other social media. *TRCM* is applied to tweet hashtags to identify evolving *ARs* at two consecutive time periods. The evolving *ARs* are then mapped to unfolding real-life topics/events reported by mainstream media. The mapping is used to demonstrate the relevance of hashtag keywords as*TDT* tool.

In chapter 1 we highlighted the benefits of hashtag label in tweets and discussed the recent increase in the reliance on Twitter for information by different entities. The challenges of detecting and tracking real-life topics/events on Twitter was also discussed in the chapter. We then proposed our research methodology for *TDT* on Twitter network. Our research aims and objectives as well as our research contributions were stated. Finally, we presented the organisation of the thesis.

In Chapter 2 we reviewed related work already carried out on Twitter data, especially those conducted in the area of *TDT*. We listed all the work reviewed and displayed the approach employed, the tools applied and the experiments conducted. It was observed that non of the techniques proposed in the literature reviewed are able to detect and track topic/event on Twitter simultaneously, hence we were movtivated to develop our proposed research methodology.

In Chapter 3 we presented an overview of Association Rule Mining *(ARM)*. We discussed *ARM* and its main concepts such as the market basket analysis, support, confidence and the lift. The two main methods for extracting *ARs* from databases namely *Apriori* and *FP-Tree* algorithms were compared and contrasted. Finally, the adoption of *Apriori* algorithm for our research experiments was justified.

In chapter 4 we introduced our research methodology termed *TRCM* and explained the adoption of the methodology for the extraction of *ARs* present in tweets' hashtags. We also discussed the measures of rule similarities and differences and present how similarity measurement is calculated. *TRCM* rules were defined with real-life examples. We explained the concept of *Rule Type Identification (RTI)* and demonstrated how tweets evolve over different Time Frame Windows (TFWs) on Twitter network. We gave real-life scenarios of each evolvement pattern. Chapter 4 was concluded by explaining the "TwO - NwO" state which demonstrates rule chain of tweet hashtags.

In chapter 5 we applied our research methodology on three real datasets from the sports and political domains.Our method was validated with previously annotated ground truth. The performance effectiveness of our experiments was calculated using *precision, recall* and the *F-Measure*. The *F-Measure* performance measurement result showed that *TRCM* performed better on the sports dataset than on those from the political domain. We attributed this to the short time-line and rapid (dynamic) evolvement of event highlights in football where a twist in event occurs within short intervals of time. Twists in political events occur less rapidly over a longer period making analysis more complex. This chapter demonstrated the capability of *TRCM* as an effective *TDT* tool for Twitter data from two different domains with varying dynamism (very rapid and less-rapid).

In chapter 6 we adopted the *Rule Type Identification Mapping (RTI-Mapping)* approach of *TRCM* to detect and track news of popular real-life events and topics. We presented qualitative case studies across different domains such as *social, political, socioeconomic* and *business.* We presented the visualisation of one of the experimental datasets using our novel **TRCM-Viz** built with *NoSQL Neo4j* to demonstrate the relationship between the hashtags and the rules (nodes). The visualisation also show "(centroids)" and their related links. Centroids demonstrated hashtags that have the most links, signifying the interestingness of such hashtags.

7.1 Reflection on Research Aims and Objectives

The research reported in this thesis focused on the extraction of hashtag keywords present in on-line tweets and the application of our novel methodology termed *TRCM* on tweet hashtags Topic Detection and Tracking *(TDT)* on Twitter. The outcome of the experiments suggests that unambiguous hashtag labelling of tweets is capable of enhancing the retrieval of topical and relevant tweets from the enormous Twitter data. The research detected newsworthy topics/events from Association Rules *ARs* embedded in tweets' at 2 consecutive periods by applying our novel methodology termed *Transaction-based Rule Change Mining (TRCM).* The experiments conducted in the research mapped hashtag keywords present in the detected *ARs* to related real-life topics/events. The experiment also visualised the relationship between evolving hashtag keywords and the *ARs* identified by *TRCM. TDT* experiments on tweets' hashtags are necessary considering the relevance of hashtag in enhancing the readability of on-line tweets, and for the description of tweets' contents. These two main important benefits of hashtag labelling on Twitter have motivated our research to empirically verify whether hashtag inclusion in topic tweets can be used to detect and track real-life topics/events in Twitter network.

7.2 Future Work

In taking the research forward, possible future work will include the following:

1. **Clustering of rules for automation of real-life news summary**. *TRCM* can be used to cluster *ARs* automatically and present detected hashtags that relates to real-life news by summarising the news headlines. The system will be used to display news and news updates headlines as they unfold. Hashtags belonging to emerging rule can be clustered and those belonging to unexpected rule can also be clustered. This can allow users the opportunity to detect and track topics/events in real-time.
2. **Comprehensive visualisation of rules to show how rules evolves in real-time**. Visualisation of all *ARs* detected in each topic/event can be displayed to show evolvement of rules in real-time using user-friendly interface. This can enable users to easily view changes in rules as the related topic/event in real-life unfolds.
3. **Application of *TRCM* to streaming tweets using *Frequent Pattern streaming* and other identified efficient techniques**. The experiments conducted in this thesis analysed downloaded tweets, however, possible future work can apply *TRCM* to Twitter streaming data. The application can be made available to Twitter users to allow them use it for day-to-day decision-making and information retrieval from Twitter enormous streaming data.

4. **Detection of resurrected rule**. In future work, a rule termed "resurrected rule" can be added to the four already identified *TRCM* rules. Resurrected rule is the rule that was initially dead on Twitter network but was re-introduced to the network due to some factors affecting its related real-life topic/event. An example of a scenario where resurrected rule can arise is a situation where a supposedly closed court case is re-visited due to new emergent evidents.

Bibliography

Adedoyin-Olowe, M., Gaber, M. M., Dancausa, C. M., and Stahl, F. (2014a). Extraction of unexpected rules from twitter hashtags and its application to sport events. In *Machine Learning and Applications (ICMLA), 2014 13th International Conference on*, pages 207–212. IEEE.

Adedoyin-Olowe, M., Gaber, M. M., and Stahl, F. (2013). Trcm: A methodology for temporal analysis of evolving concepts in twitter. In *Artificial Intelligence and Soft Computing*, pages 135–145. Springer.

Adedoyin-Olowe, M., Gaber, M. M., and Stahl, F. (2014b). A Survey of Data Mining Techniques for Social Media Analysis. *Journal of Data Mining & Digital Humanities*, 2014.

Adedoyin-Olowe, M., Gaber, M. M., Stahl, F., and Gomes, J. B. (2015). Autonomic discovery of news evolvement in twitter. In *Big Data in Complex Systems*, pages 205–229. Springer.

Agarwal, P., Vaithiyanathan, R., Sharma, S., and Shroff, G. (2012). Catching the long-tail: Extracting local news events from twitter. In *proceedings of Sixth International AAAI Conference on Weblogs and Social Media*, 4 - 7 June 2012, Dublin, pages 379 – 382. ICWSM.

Aggarwal, C. C. (2011). *An introduction to social network data analytics.* Springer.

Agichtein, E., Castillo, C., Donato, D., Gionis, A., and Mishne, G. (2008). Finding high-quality content in social media. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 183–194. ACM.

Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. In *ACM SIGMOD Record*, volume 22, pages 207–216. ACM.

Agrawal, R. and Srikant, R. (1995). Mining sequential patterns. In *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*, pages 3–14. IEEE.

Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499.

Aiello, L. M., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., Goker, A., Kompatsiaris, I., and Jaimes, A. (2013). Sensing trending topics in twitter. *IEEE Transactions on, 15(6), 1268-1282.*

Ale, J. M. and Rossi, G. H. (2000). An approach to discovering temporal association rules. In *Proceedings of the 2000 ACM symposium on Applied computing-Volume 1*, pages 294–300. ACM.

Allan, J. (2002a). Introduction to topic detection and tracking. In *Topic detection and tracking*, pages 1–16. (pp. 1-16). Springer US.

Allan, J. (2002b). *Topic detection and tracking: event-based information organization*, volume 12. Springer.

Allan, J., Carbonell, J. G., Doddington, G., Yamron, J., and Yang, Y. (1998). Topic detection and tracking pilot study final report.

Anjaria, M. and Guddeti, R. M. R. (2014). Influence factor based opinion mining of twitter data using supervised learning. In *COMSNETS*, pages 1–8.

Ausserhofer, J. and Maireder, A. (2013). National politics on twitter: structures and topics of a networked public sphere. *Information, Communication & Society*, 16(3):291–314.

Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval*, volume 463. ACM press New York.

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.

Balakrishnan, R. and Ranganathan, K. (2012). *A textbook of graph theory*. Springer Science & Business Media.

Bayardo Jr, R. J. and Agrawal, R. (1999). Mining the most interesting rules. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 145–154. ACM.

Becker, H., Iter, D., Naaman, M., and Gravano, L. (2012). Identifying content for planned events across social media sites. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 533–542. ACM.

Becker, H., Naaman, M., and Gravano, L. (2011). Beyond trending topics: Real-world event identification on twitter. *ICWSM*, 11:438–441.

Benhardus, J. and Kalita, J. (2013). Streaming trend detection in twitter. *International Journal of Web Based Communities*, 9(1):122–139.

Bifet, A. and Frank, E. (2010). Sentiment knowledge discovery in twitter streaming data. In *Discovery Science*, pages 1–15. Springer.

Bizer, C., Boncz, P., Brodie, M. L., and Erling, O. (2012). The meaningful use of big data: four perspectives–four challenges. *ACM SIGMOD Record*, 40(4):56–60.

Bollen, J., Mao, H., and Pepe, A. (2011a). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *ICWSM*.

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

Bollier, D. and Firestone, C. M. (2010). *The promise and peril of big data*. Aspen Institute, Communications and Society Program Washington, DC, USA.

Bramer, M. (2013). *Principles of Data Mining*. Springer Science & Business Media.

Bramer, M., Bramer, M., and Bramer, M. (2007). *Principles of data mining*, volume 180. Springer.

Brin, S., Motwani, R., and Silverstein, C. (1997a). Beyond market baskets: Generalizing association rules to correlations. In *ACM SIGMOD Record*, volume 26, pages 265–276. ACM.

Brin, S., Motwani, R., Ullman, J. D., and Tsur, S. (1997b). Dynamic itemset counting and implication rules for market basket data. In *ACM SIGMOD Record*, volume 26, pages 255–264. ACM.

Castillo, C., Mendoza, M., and Poblete, B. (2011). Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684. ACM.

Cataldi, M., Di Caro, L., and Schifanella, C. (2010). Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, page 4. ACM.

Chakrabarti, D. and Punera, K. (2011). Event summarization using tweets. In *proceedings of the 5th international conference on weblogs and social media*, 17 - 21 July 2011, Barcelona, pages 66–73, ICWSM.

Chang, H.-C. (2010). A new perspective on twitter hashtag use: diffusion of innovation theory. *Proceedings of the American Society for Information Science and Technology*, 47(1):1–4.

Chen, W.-K. (2012). *Applied graph theory*, volume 13. Elsevier.

Chen, Y., Amiri, H., Li, Z., and Chua, T.-S. (2013). Emerging topic detection for organizations from microblogs. *In Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval (pp. 43-52). ACM.*

Cheong, M. and Lee, V. (2009). Integrating web-based intelligence retrieval and decision-making from the twitter trends knowledge base. In *Proceedings of the 2nd ACM workshop on Social web search and mining*, pages 1–8. ACM.

Conover, M. D., Gonçalves, B., Ratkiewicz, J., Flammini, A., and Menczer, F. (2011). Predicting the political alignment of twitter users. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom)*, pages 192–199. IEEE.

Corney, D., Martin, C., and Göker, A. (2014). Spot the ball: Detecting sports events on twitter. In *Advances in Information Retrieval*, pages 449–454. Springer.

Deo, N. (2004). *Graph theory with applications to engineering and computer science.* PHI Learning Pvt. Ltd.

Diakopoulos, N. A. and Shamma, D. A. (2010). Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1195–1198. ACM.

Dong, A., Zhang, R., Kolari, P., Bai, J., Zheng, F. D. Y. C. Z., and Zha, H. (2010). Time is of the essence: Improving recency ranking using twitter data. *Proceedings of the 19th international conference on World wide web Pages 331-340.*

Dong, G. and Li, J. (1999). Efficient mining of emerging patterns: Discovering trends and differences. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 43–52. ACM.

Efron, M. (2010). Hashtag retrieval in a microblogging environment. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 787–788. ACM.

Elvers, T. and Srinivasan, P. (2011). What's trending?: mining topical trends in ugc systems with youtube as a case study. In *Proceedings of the Eleventh International Workshop on Multimedia Data Mining*, page 4. ACM.

Evans, D. (2010). *Social media marketing: the next generation of business engagement*. John Wiley & Sons.

Feng, W. and Wang, J. (2014). We can learn your # hashtags: Connecting tweets to explicit topics. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 856–867. IEEE.

Fukuda, T., Morimoto, Y., Morishita, S., and Tokuyama, T. (1996). Data mining using two-dimensional optimized association rules: Scheme, algorithms, and visualization. *ACM SIGMOD Record*, 25(2):13–23.

GENG, L. and HAMILTON, H. J. (2006). Interestingness measures for data mining: A survey. *ACM Computing Surveys (CSUR), 38(3), 9.*

Glass, K. and Colbaugh, R. (2010). Toward emerging topic detection for business intelligence: Predictive analysis ofmeme'dynamics. Technical report, arXiv preprint arXiv:1012.5994.

Gomes, J. B., Adedoyin-Olowe, M., Gaber, M. M., and Stahl, F. (2013). Rule type identification using trcm for trend analysis in twitter. In *Research and Development in Intelligent Systems XXX*, pages 273–278. Springer.

Grosseck, G. and Holotescu, C. (2008). Can we use twitter for educational activities. In *4th international scientific conference, eLearning and software for education, Bucharest,*, 17 - 18 April 2008, Bucharest.

Guzman, J. and Poblete, B. (2013). On-line relevant anomaly detection in the twitter stream: an efficient bursty keyword detection model. In *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description*, pages 31–39. ACM.

Hahsler, M. and Chelluboina, S. (2011). Visualizing association rules in hierarchical groups. In *42nd Symposium on the Interface: Statistical, Machine Learning, and Visualization Algorithms (Interface 2011). The Interface Foundation of North America.*

Han, J., Dong, G., and Yin, Y. (1999). Efficient mining of partial periodic patterns in time series database. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 106–115. IEEE.

Han, J., Pei, J., and Yin, Y. (2000). Mining frequent patterns without candidate generation. In *ACM SIGMOD Record*, volume 29, pages 1–12. ACM.

Hand, D. J., Mannila, H., and Smyth, P. (2001). *Principles of data mining.* MIT press.

Henno, J., Jaakkola, H., Mäkelä, J., and Brumen, B. (2013). Will universities and university teachers become extinct in our bright online future? In *Information & Communication Technology Electronics & Microelectronics (MIPRO), 2013 36th International Convention on*, pages 716–725. IEEE.

Hipp, J., Güntzer, U., and Nakhaeizadeh, G. (2000). Algorithms for association rule mining a general survey and comparison. *ACM sigkdd explorations newsletter*, 2(1):58–64.

Holzschuher, F. and Peinl, R. (2013). Performance of graph query languages: comparison of cypher, gremlin and native access in neo4j. In *Proceedings of the Joint EDBT/ICDT 2013 Workshops*, pages 195–204. ACM.

Indyk, P. and Motwani, R. (1998). Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 604–613. ACM.

Inouye, D. and Kalita, J. K. (2011). Comparing twitter summarization algorithms for multiple post summaries. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 298–306. IEEE.

Jackoway, A., Samet, H., and Sankaranarayanan, J. (2011). Identification of live news events using twitter. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks*, pages 25–32. ACM.

Jain, D., Khatri, P., Soni, R., and Chaurasia, B. K. (2012). Hiding sensitive association rules without altering the support of sensitive item (s). In *Advances in Computer Science and Information Technology. Networks and Communications*, pages 500–509. Springer.

Jiawei Han, Micheline Kamber, J. P. P. (2011). *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc.

Jin, R., Yang, G., and Agrawal, G. (2005). Shared memory parallelization of data mining algorithms: Techniques, programming interface, and performance. *Knowledge and Data Engineering, IEEE Transactions on*, 17(1):71–89.

Joshi, A. and Sodhi, J. (2014). Target advertising via association rule mining. *International Journal*, 2(5).

Kaplan, A. M. (2012). If you love something, let it go mobile: Mobile marketing and mobile social media 4x4. *Business Horizons*, 55(2):129–139.

Kaplan, A. M. and Haenlein, M. (2009). The fairyland of second life: Virtual social worlds and how to use them. *Business horizons*, 52(6):563–572.

Kasiviswanathan, S. P., Melville, P., Banerjee, A., and Sindhwani, V. (2011). Emerging topic detection using dictionary learning. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 745–754. ACM.

Kleinberg, J. (2003). Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397.

Kotsiantis, S. and Kanellopoulos, D. (2006). Association rules mining: A recent overview. *GESTS International Transactions on Computer Science and Engineering*, 32(1):71–82.

Kouloumpis, E., Wilson, T., and Moore, J. (2011). Twitter sentiment analysis: The good the bad and the omg! *ICWSM*, 11:538–541.

Krishnamurthy, B., Gill, P., and Arlitt, M. (2008). A few chirps about twitter. In *Proceedings of the first workshop on Online social networks*, pages 19–24. ACM.

Kumar, B. S. and Rukmani, K. (2010). Implementation of web usage mining using apriori and fp growth algorithms. *Int. J. of Advanced Networking and Applications*, 1(06):400–404.

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.

Laniado, D. and Mika, P. (2010). Making sense of twitter. In *The Semantic Web–ISWC 2010*, pages 470–485. Springer.

Lasorsa, D. L., Lewis, S. C., and Holton, A. E. (2012). Normalizing twitter: Journalism practice in an emerging communication space. *Journalism Studies*, 13(1):19–36.

Lau, J. H., Collier, N., and Baldwin, T. (2012). On-line trend analysis with topic models:\# twitter trends detection topic model online. In *COLING*, pages 1519–1534.

Lent, B., Swami, A., and Widom, J. (1997). Clustering association rules. In *Data Engineering, 1997. Proceedings. 13th International Conference on*, pages 220–231. IEEE.

Lerman, K. and Ghosh, R. (2010). Information contagion: An empirical study of the spread of news on digg and twitter social networks. *ICWSM*, 10:90–97.

Levenberg, A. and Osborne, M. (2009). Stream-based randomised language models for smt. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 756–764. Association for Computational Linguistics.

Li, J., Vishwanath, A., and Rao, H. R. (2014). Retweeting the fukushima nuclear radiation disaster. *Communications of the ACM*, 57(1):78–85.

Li, N., Zeng, L., He, Q., and Shi, Z. (2012). Parallel implementation of apriori algorithm based on mapreduce. In *Software Engineering, Artificial Intelligence, Networking and Parallel & Distributed Computing (SNPD), 2012 13th ACIS International Conference on*, pages 236–241. IEEE.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Liu, B., Hsu, W., and Ma, Y. (1999). Mining association rules with multiple minimum supports. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 337–341. ACM.

Liu, D.-R., Shih, M.-J., Liau, C.-J., and Lai, C.-H. (2009). Mining the change of event trends for decision support in environmental scanning. *Expert Systems with Applications*, 36(2):972–984.

Liu, G., Suchitra, A., Zhang, H., Feng, M., Ng, S.-K., and Wong, L. (2012). Assocexplorer: an association rule visualization system for exploratory data analysis. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1536–1539. ACM.

Long, G. V. D. D. and Yu, J. X. (2009). Web information systems engineering–wise 2009.

Mannila, H., Toivonen, H., and Verkamo, A. I. (1997). Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289.

Mathioudakis, M. and Koudas, N. (2010). Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158. ACM.

McCreadie, R., Macdonald, C., Ounis, I., Osborne, M., and Petrovic, S. (2013). Scalable distributed event detection for twitter. In *Big Data, 2013 IEEE International Conference on*, pages 543–549. IEEE.

McGarry, K. (2005). A survey of interestingness measures for knowledge discovery. *Knowledge Eng. Review*, 20(1):39–61.

Meng, X., Wei, F., Liu, X., Zhou, M., Li, S., and Wang, H. (2012). Entity-centric topic-oriented opinion summarization in twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 379–387. ACM.

Meyer, B., Bryan, K., Santos, Y., and Kim, B. (2011). Twitterreporter: Breaking news detection and visualization through the geo-tagged twitter network. In *CATA*, pages 84–89.

Naaman, M., Becker, H., and Gravano, L. (2011). Hip and trendy: Characterizing emerging trends on twitter. *Journal of the American Society for Information Science and Technology*, 62(5):902–918.

Naaman, M., Boase, J., and Lai, C.-H. (2010). Is it really about me?: message content in social awareness streams. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 189–192. ACM.

Newman, N. (2009). The rise of social media and its impact on mainstream journalism. *Reuters Institute for the Study of Journalism*.

Ong, K.-H., Ong, K.-L., Ng, W.-K., and Lim, E.-P. (2002). Crystalclear: Active visualization of association rules. In *ICDM-02 Workshop on Active Mining (AM-02)*. Citeseer.

Osborne, M., Petrovic, S., McCreadie, R., Macdonald, C., and Ounis, I. (2012). Bieber no more: First story detection using twitter and wikipedia. In *Proceedings of the Workshop on Time-aware Information Access. TAIA*, volume 12.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of Seventh International Conference on Language Resources and Evaluation, LREC*, 17 - 23 May 2010, Malta, pages 1320 – 1326. LREC.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Parameswaran, M. and Whinston, A. B. (2007). Social computing: An overview. *Communications of the Association for Information Systems*, 19(1), 37.

Park, J. S., Chen, M.-S., and Yu, P. S. (1997). Using a hash-based method with transaction trimming for mining association rules. *Knowledge and Data Engineering, IEEE Transactions on*, 9(5):813–825.

Petrović, S., Osborne, M., and Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189. Association for Computational Linguistics.

Petrovic, S., Osborne, M., McCreadie, R., Macdonald, C., Ounis, I., and Shrimpton, L. (2013). Can twitter replace newswire for breaking news? In *ICWSM*.

Phuvipadawat, S. and Murata, T. (2010). Breaking news detection and tracking in twitter. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, volume 3, pages 120–123. IEEE.

Popescu, A.-M. and Pennacchiotti, M. (2010). Detecting controversial events from twitter. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1873–1876. ACM.

Prangsma, M. E., Boxtel, C. A., Kanselaar, G., and Kirschner, P. A. (2009). Concrete and abstract visualizations in history learning tasks. *British Journal of Educational Psychology*, 79(2):371–387.

Qualman, E. (2012). *Socialnomics: How social media transforms the way we live and do business.* John Wiley & Sons.

Read, J. (2005). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, pages 43–48. Association for Computational Linguistics.

Ritter, A., Etzioni, O., Clark, S., et al. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1104–1112. ACM.

Safko, L. (2010). *The Social media bible: tactics, tools, and strategies for business success*. John Wiley & Sons.

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.

Sayyadi, H., Hurst, M., and Maykov, A. (2009). Event detection and tracking in social streams. In *ICWSM*.

Shamma, D. A., Kennedy, L., and Churchill, E. F. (2010). Summarizing media through short-messaging services. In *Proceedings of the ACM conference on computer supported cooperative work*, 6 - 10 Fabruary 2010, Savannah. Citeseer.

Shi, Q., Petterson, J., Dror, G., Langford, J., Strehl, A. L., Smola, A. J., and Vishwanathan, S. (2009). Hash kernels. In *International Conference on Artificial Intelligence and Statistics*, pages 496–503.

Shirky, C. (2004). Blog explosion and insiders club: Brothers in cluelessness. Electronic.

Silverstein, C., Brin, S., Motwani, R., and Ullman, J. (2000). Scalable techniques for mining causal structures. *Data Mining and Knowledge Discovery*, 4(2-3):163–192.

Singh, J., Ram, H., and Sodhi, D. J. (2013). Improving efficiency of apriori algorithm using transaction reduction. *International Journal of Scientific and Research Publications*, 3(1):1–4.

Song, H. S., Kim, S. H., et al. (2001). Mining the change of customer behavior in an internet shopping mall. *Expert Systems with Applications*, 21(3):157–168.

Srikant, R. and Agrawal, R. (1996). Mining quantitative association rules in large relational tables. In *ACM SIGMOD Record*, volume 25, pages 1–12. ACM.

Srikant, R., Vu, Q., and Agrawal, R. (1997). Mining association rules with item constraints. In *KDD*, volume 97, pages 67–73.

Starbird, K. and Palen, L. (2012). (how) will the revolution be retweeted?: information diffusion and the 2011 egyptian uprising. In *Proceedings of the acm 2012 conference on computer supported cooperative work*, pages 7–16. ACM.

Takahashi, T., Tomioka, R., and Yamanishi, K. (2011). Discovering emerging topics in social streams via link anomaly detection. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 1230–1235. IEEE.

Tjioe, H. C. and Taniar, D. (2004). A framework for mining association rules in data warehouses. In *Intelligent Data Engineering and Automated Learning–IDEAL 2004*, pages 159–165. Springer.

Tomokiyo, T. and Hurst, M. (2003). A language model approach to keyphrase extraction. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18*, pages 33–40. Association for Computational Linguistics.

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*, 10:178–185.

van Oorschot, G., van Erp, M., and Dijkshoorn, C. (2012). Automatic extraction of soccer game events from twitter. In *Proc. of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web.*

Verma, S., Vieweg, S., Corvey, W. J., Palen, L., Martin, J. H., Palmer, M., Schram, A., and Anderson, K. M. (2011). Natural language processing to the rescue? extracting" situational awareness" tweets during mass emergency. In *ICWSM*. Citeseer.

Vieweg, S., Hughes, A. L., Starbird, K., and Palen, L. (2010). Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1079–1088. ACM.

Walther, M. and Kaisser, M. (2013). Geo-spatial event detection in the twitter stream. In *Advances in Information Retrieval*, pages 356–367. Springer.

Wang, H., Can, D., Kazemzadeh, A., Bar, F., and Narayanan, S. (2012). A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations*, pages 115–120. Association for Computational Linguistics.

Watanabe, K., Ochi, M., Okabe, M., and Onai, R. (2011). Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2541–2544. ACM.

Webber, J. (2012). A programmatic introduction to neo4j. In *Proceedings of the 3rd annual conference on Systems, Programming, and Applications: Software for Humanity*, pages 217–218. ACM.

Weng, J. and Lee, B.-S. (2011). Event detection in twitter. In *the proceedings of the international conference on weblogs and social media*, 17 - 21 July 2011, Barcelona, pages 401 – 408. ICWSM.

Whiting, S., Zhou, K., Jose, J., Alonso, O., and Leelanupab, T. (2012). Crowdtiles: presenting crowd-based information for event-driven information needs. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2698–2700. ACM.

Wong, P. C., Whitney, P., and Thomas, J. (1999). Visualizing association rules for text mining. In *Information Visualization, 1999.(Info Vis' 99) Proceedings. 1999 IEEE Symposium on*, pages 120–123. IEEE.

Yang, C., Lin, K. H., and Chen, H.-H. (2007). Emotion classification using web blog corpora. In *Web Intelligence, IEEE/WIC/ACM International Conference on*, pages 275–278. IEEE.

Yang, J. and Honavar, V. (1998). Feature subset selection using a genetic algorithm. In *Feature extraction, construction and selection*, pages 117–136. Springer.

Yao, L., Mimno, D., and McCallum, A. (2009). Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 937–946. ACM.

Yardi, S., Romero, D., Schoenebeck, G., et al. (2009). Detecting spam in a twitter network. *First Monday*, 15(1).

Zaki, M. J. and Hsiao, C.-J. (2002). Charm: An efficient algorithm for closed itemset mining. In *SDM*, volume 2, pages 457–473. SIAM.

Zhang, Y., Tsai, F. S., and Kwee, A. T. (2011). Multilingual sentence categorization and novelty mining. *Information Processing & Management*, 47(5):667–675.