Real-time Content Identification for Events and Sub-Events from Microblogs.

Wang, Xinyue

QUEEN MARY UNIVERSITY OF LONDON

School of Electronic Engineering and Computer Science

Real-time Content Identification for Events and Sub-Events from Microblogs

by **Xinyue Wang**

Feb 15, 2016

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

Statement of originality

I, Xinyue Wang, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

Signature:Xinyue Wang

Date: February 15, 2016

QUEEN MARY UNIVERSITY OF LONDON
SCHOOL OF ELECTRONIC ENGINEERING AND COMPUTER SCIENCE

ABSTRACT

Real-time Content Identification for Events and Sub-Events from Microblogs

by Xinyue Wang

In an age when people are predisposed to report real-world events through their social media accounts, many researchers value the advantages of mining such unstructured and informal data from social media. Compared with the traditional news media, online social media services, such as Twitter, can provide more comprehensive and timely information about real-world events. Existing Twitter event monitoring systems analyse partial event data and are unable to report the underlying stories or sub-events in real-time. To fill this gap, this research focuses on the automatic identification of content for events and sub-events through the analysis of Twitter streams in real-time.

To fulfil the need of real-time content identification for events and sub-events, this research first proposes a novel *adaptive crawling* model that retrieves extra event content from the Twitter Streaming API. The proposed model analyses the characteristics of hashtags and tweets collected from live Twitter streams to automate the expansion of subsequent queries. By investigating the characteristics of Twitter hashtags, this research then proposes three Keyword Adaptation Algorithms (KwAAs) which are based on the term frequency (TF-KwAA), the traffic pattern (TP-KwAA), and the text content of associated tweets (CS-KwAA) of the emerging hashtags. Based on the comparison between traditional keyword crawling and adaptive crawling with different KwAAs, this thesis demonstrates that the KwAAs retrieve extra event content about sub-events in real-time for both planned and unplanned events.

To examine the usefulness of extra event content for the event monitoring system, a Twitter event monitoring solution is proposed. This "*Detection of Sub-events by Twitter Real-time Monitoring (DSTReaM)*" framework concurrently runs multiple instances of a statistical-based event detection algorithm over different stream components. By evaluating the detection performance using detection accuracy and event entropy, this research demonstrates that better event detection can be achieved with a broader coverage of event content.

Contents

List of Figures

List of Tables

List of Abbreviations

TDT	Topic Detection and Tracking
KwAAs	Keyword Adaptation Algorithms
TF-KwAA	Term Frequency based Keyword Adaptation Algorithm
TP	Traffic Pattern based Keyword Adaptation Algorithm
CS	Content Similarity based Keyword Adaptation Algorithm
TT	Topic Tracking
TD	Topic Detection
FSD	First Story Detection
API	Application Program Interface
TEM	Twitter Event Monitoring
NED	New Event Detection
RED	Retrospective Event Detection
PRF	Pseudo Relevance Feedback
TF-IDF	Term Frequency - Inverse Document Frequency
EWMA	Exponentially Weighted Moving Average
MH17	Malaysia Airlines Flight 17
BL	Baseline crawler/dataset
AD	Adaptive crawler/dataset
EX	Extra dataset
p	Precision
r	Recall
F_1	F1 score
DSTReam	Detection of Sub-events by Twitter Real-time Monitoring

Acknowledgements

First and foremost, I would like to express my sincere gratitude to Dr. Laurissa Tokarchuk and Dr. Stefan Poslad for their supervision, advice, understanding and support throughput my whole Ph.D. Laurissa's patient guidance of research and unflagging enthusiasm towards life have inspired me a lot. Stefan always encouraged me to think from a systematic viewpoint toward my research and gave a lot of guidance on technical and logical writing. I deeply appreciate their contribution of time and work to make this research stimulating. Thank you both!

I would also like to acknowledge the financial support from School of Electronic Engineering and Computer Science (EECS), Queen Mary University of London (QMUL) and China Scholarship Council (CSC), who enabled this research to be conducted. I also wish to thank Dr Félix Cuadrado for the technical support he gave.

Additionally, I am also very grateful for the support from my friends, and other members of the Network Research Group and Cognitive Science Research Group. They have brought great pleasure to light up my life and it is great to have them around.

Finally, special thanks must go to my beloved family. They are always ready to give me support and love. And also to my good friend Teng Jiang, I thank him for the tolerance and understanding of my occasional bad mood. Without his unwavering support and help, I wouldn't be as able to survive the hard times.

To My Family

Author's Publication

- Xinyue Wang, Laurissa Tokarchuk, Félix Cuadrado, and Stefan Poslad. Adaptive Identification of Hashtags for Real-time Event Data Collection. In: Recommendation and Search in Social Networks. Eds. Ulusoy, O., Tansel A.U., and Arkun, E., Lecture Notes in Social Network, 2015, 1-22

- Xinyue Wang, Laurissa Tokarchuk, and Stefan Poslad. Identifying Relevant Event Content for Real-time Event Detection. IEEE/ACM. International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), Beijing, China, August 2014, 395 - 398

- Xinyue Wang, Laurissa Tokarchuk, Flix Cuadrado, and Stefan Poslad. Exploiting Hashtags for Adaptive Microblog Crawling. IEEE/ACM International Conference on Social Networks Analysis and Mining (ASONAM 2013), Ontario, Canada, August 2013, 311-315

- Andrea Zielinski, Stuart E. Middleton, Laurissa Tokarchuk and Xinyue Wang. Social-media Text Mining and Network Analysis to support Decision Support for Natural Crisis Management. International Conference on Information Systems for Crisis Response and Management (ISCRAM 2013)

Chapter 1

Introduction

The desire for knowledge of both planned and unplanned events in part drives the thirst to consume news about current events. What in part drives our thirst for knowledge about unplanned events is simply to find out more information about unforeseen things. What in part drives our thirst for knowledge about planned events is the need to know about the underlying stories of the events: such as if there is a sports-event, what is the progress of specific sports players we are interested in. Since the issue of the first newspaper in 1605, traditional news media, including the printed media, broadcast media and newswires, acts as the principle channel for the general public to get knowledge about events in the world around them. With the huge explosion of different news sources about worldwide events, it can be overwhelming for readers to manually find stories of events that interest them. As a result, the Topic Detection and Tracking (TDT) project was initiated. It provides ways to automatically identify specific sets of stories that relate to newsworthy events [1]. All of these solutions rely on text retrieval and clustering techniques to detect events from a temporal-ordered structured text stream [2, 3].

However, the traditional news media is characterised by "one-way" news consumption: readers are only allowed to be passively involved in the process. Since events news are packaged by media professionals, people's knowledge of such reported events can be misrepresented, e.g. even professional news reports can be biased with personal opinions. The traditional news media tends to mask the ability of the general public to express

	BRA	SPA	DEN	ITA	US	FIN	UK	FRA	GER	JPN	AUS	IRE
TV	81%	82%	75%	78%	64%	75%	75%	80%	82%	73%	72%	76%
Radio	39%	40%	50%	23%	26%	45%	37%	28%	50%	17%	41%	50%
Printed Newspapers	33%	47%	49%	38%	23%	49%	38%	19%	38%	44%	39%	49%
Online (inc. social media)	91%	86%	83%	81%	74%	90%	73%	71%	60%	70%	85%	83%

FIGURE 1.1: Source of News by Country. Proportion of news consumers that used the particular sources of news in the last week (reproduced from [8])

their own opinions, and also prevents media consumers from obtaining a comprehensive overview of events [4]. Event news is often time and location-sensitive, journalists may only have an incomplete or partial view of the event when they arrive on the scene. Moreover, the lengthy *production pipeline* followed by the traditional news media for generating news reports (acquiring, writing, reporting and producing) increases the reporting latency [5]. Although the emerging new media, i.e. using online portals of news agency, alleviates some of the issues in traditional news media through a more open and easier accessed infrastructure, the perception of news is still affected by the "one-way" communication pattern and the time consuming production pipeline.

Starting with Web 1.0 that allowed anyone to publish a blog and interlink it to other information, Web 2.0 has continued to reshape the way people interact with the rest of the world and engage in events. Instead of passively consuming online news as a reader, the general public is drawn in to contribute their ideas through Web 2.0 applications [6]. As a representative type of Web 2.0 application, social media not only blurs the line between information disseminators and receivers, but also breaks the barriers between news experts and amateurs [7]. This more social type of news consumption raises unprecedented challenges to traditional news media and online newswire. Nowadays, a large proportion of the population seeks to use online media for news (as shown in Figure 1.1), and the amount of people who rely on social media to find news is increasing (as shown in Figure 1.2).

By encouraging the general public to report their observations and to express their opinions about real-world events, online social media provides a more open and flexible platform for sharing events information. Equipped with an increasingly influential army

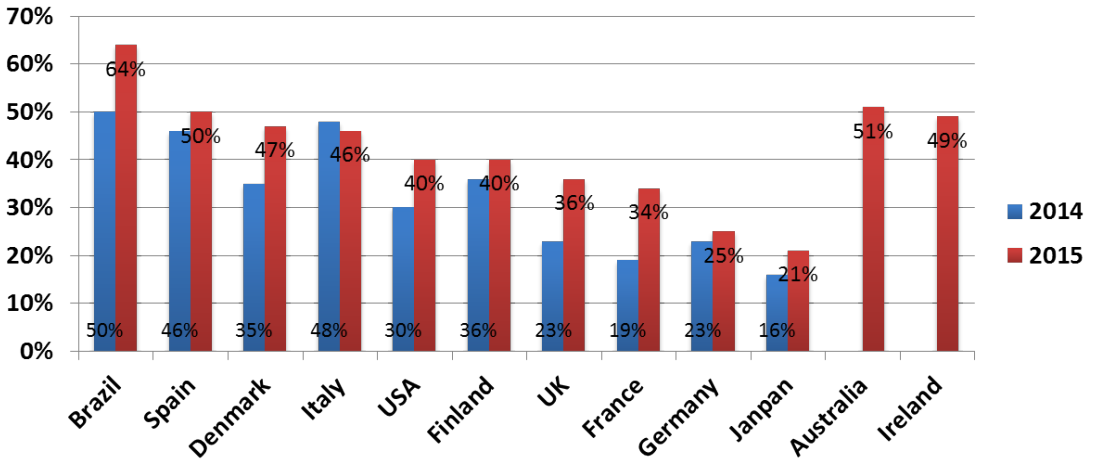


FIGURE 1.2: Using Social Media as the Source of News. Proportion of news consumer that used social media as the sources of news in 2013 and 2015 (reproduced from [8])

of "citizen journalists" [9] and "social sensors" [10], online social media are becoming the microphone and camera for mass events [11]. Many real-world examples can be used to illustrate the effectiveness of online social media during news events. For instance, warning messages during Virginia Tech shooting in April 2007 came primarily from students and unofficial sources via the online social media [12]; the devastating bomb blasts in Mumbai in November 2008 also relied on online social media for decision making [13]; one of the most well-known example is the "Arab Spring"": social media became the primary medium to unravel the progress of this revolution [14, 15]. Additionally, in contrast to the long latency (days, weeks or even months) of traditional news media, social media provides quicker access and more comprehensive information [16]. Monitoring and analysing this fruitful and dynamic flow of people's reports can yield precious information, which would not have been available from traditional media outlets [17]. Consequently, researchers today start to value the advantages of mining large scale data from social media [18, 19, 20].

1.1 Motivation and Challenges

Microblogging services, such as Twitter[1], are becoming a prominent communication tool for news dissemination [21, 22]. Twitter supports this by the concise expression of ideas and opinions via a tweet, i.e. a short text in format limited to 140 characters. Users can instantly post and access tweets about the latest local and worldwide news. Among the more than 300 million active users [23], 56% of them post tweets about current events in real-time [24]. Empirical studies demonstrate that Twitter not only reveals the broadcast events [25, 26], but also becomes the preferred medium for discovering breaking news [27, 28].

Although the less strict requirement on content quality facilitates the adoption of Twitter, this feature also infuses its content with a great amount of noise information. People share not only their comments about events, but also about their most trivial matters of daily lives. As a result, Twitter streams contain large amounts of meaningless chats [29], advertisements [30] and even rumours [31]. In the last decade, notable research efforts have been made to try to distinguish the informative tweets about real-world events from the rest of the background noise information [32, 33, 34, 35, 36, 37, 38]. The main focus is to quickly identify the events from the mass Twitter stream. However, rather than the efficiency of reporting the breaking news ahead of online newswires [27], the advantages of Twitter lies in its effectiveness as follows. Twitter offers a broader coverage of event information intertwined with additional viewpoints [39] and the capability in revealing wider aspects about the evolution of events [40]. By scrolling through the Twitter timeline and tracking live Twitter streams, it is possible to acquire information about both the events and the underlying stories (or sub-events) for a fuller picture of an event.

To monitor the finer granularities, i.e. sub-events, of an event, common practice is to analyse filtered Twitter streams using event detection approaches that are developed for conventional TDT tasks. A large proportion of these solutions detect different phases of disaster events for offering situation awareness to both general public and government

[1]Twitter: `https://twitter.com`

authorities [41, 42, 43, 44]. Other research work investigates the sub-events of planned events, such as sport competitions [45, 46, 47, 48], festival activities [49, 50] and political elections [51, 52, 53]. However, their solutions still have the following drawbacks:

- **Use pre-defined search criteria**

  A common assumption made by most existing research is that all sub-event events information is retrievable using pre-defined and constant keywords. Therefore, their conclusions are based on static datasets that represents the status of the event at a particular time point. However, pre-defined keywords are subjective and new topics about sub-events often arise in the midst of events [112]. It is necessary to expand the coverage of event information during the event.

- **Infeasible to run in real-time**

  To better detect and understand sub-events, researchers upgrade the existing solutions by running them in multiple iterations. This is at the cost of extra complexity and additional resources. Existing research has shown that the traditional event detection algorithms don't scale to huge volume of high speed streaming data, such as tweets [32]. Since a main feature of Twitter is to provide instantaneous access to the event information, a solution that is capable to get timely event information with good accuracy is desired.

- **Focus on single type of events**

  Some existing solutions are based on the assumption that the prior knowledge of the events exists [49, 50]. Although external resources can provide a priori knowledge of planned events, this is not the case for unplanned events, such as protests and crises. Unplanned events are by their nature unanticipated and hard to discover or predict. Sub-event monitoring of unplanned events are normally designed to meet the information needs during aftermath. These solutions are event specific and have strict requirements for the input data [44, 54]. Since the existing solutions tend to focus on a single type of events, hence, information analysis that can cope with both planned and unplanned events is required.

Based on the above analysis, a research gap exists in supporting real-time event monitor-

ing with information about the underlying stories and subsequent events. Consequently, the focus of this research is to provide a fuller picture of breaking news events by detecting and summarising the underlying meaningful sub-events in real-time. To achieve this research aim, some essential challenges raised by both Twitter's characteristics and an event's characteristics need to be solved:

- **The short-length nature of tweets.** Admittedly, event information from Twitter is easier to consume and faster to spread due to this feature. However, compared with conventional documents, the apparent reduction in document length of tweets can be problematic. Commonly, a text document is modelled with the probability distribution of its term, such as the bag of word model [55] or topic model [56]. Short text documents such as tweets thus can thus result in term sparsity issue and become incompatible with existing text mining techniques.

- **The high arrival rate of tweets.** The main reason for analysing events through Twitter streams is its ability to reveal the evolution of events in real-time. However, the velocity and volume of tweets produced in every single second is continuously growing [57]. While the existing systems are designed to deal with a reduced corpus (i.e. tweets are preprocessed with noises filtering), they are unable to scale to a large amount of tweets [58].

- **Noisy tweets in Twitter events stream.** Acquisition of event-relevant Twitter posts from the noisy Microblog environments can be a non-trivial task. Although the focus of this research is to discover information about sub-events from a text stream about a certain event, the background noise in the diverse and poor quality tweets still needs to be considered. This is critical if better quality information about the underlying stories needs to be provided.

- **The diversity across various types of events.** Different type of events are described and discussed with a different vocabulary. Even for similar events that share common terms (e.g. FIFA World Cup[2] and the Football competition of

[2]an international association football competition, held every four years, contested by the senior men's national teams of the members of Fédération Internationale de Football Association (FIFA)

Olympic Games), the amount of tweets traffic associated with each of them varies. In fact, the information about any two events differ significantly in content, number of messages and participants, periods, inherent structure, and causal relationships [59], thus making the idea of a one-fit-all solution, seem unlikely.

1.2 Research Objectives

In order to detect and summarise sub-events and subsequent stories of the events , this research will build on existing event detection research. By collecting, detecting and then extracting the event-relevant tweets, the final output is exploited to formulate the overview of the events. Whilst the existing research focuses on the depth of detection, i.e. on more accurate detection results with sophisticated but inefficient algorithms, the focus of this research is in achieving the same goal by increasing the coverage of the event content.

Specifically, the main objective of this research can be stated as: ***to provide a better event monitoring solution that identifies the newsworthy sub-events in an online manner by exploiting the expanded coverage of online social media text documents, e.g. tweets about the event of interest.*** This research fulfils the main objective by achieving the following three sub-objectives:

1. to explore whether there exists extra event content in addition to the datasets used by the existing solutions and to design a microblog crawling model that enables extra event-relevant content to be collected in real-time;

2. to identify features or metrics that can be used to retrieve event-relevant contents and to relate these features to the Twitter crawling model;

3. to investigate the performance of sub-event detection with a broader coverage of event content by developing a new event monitoring solution that incorporates a new Twitter crawling model.

1.3 Contributions and Novelty

This research proposes to improve the Twitter event monitoring system by automatically mining a comprehensive set of event content based on live streaming tweets. The main, novel, contributions of this thesis are as follows:

1. A novel model of real-time event content retrieval for streaming text, called *adaptive microblog crawling*, or simply *adaptive crawling*. This model analyses the characteristics of incoming Twitter streams in real-time to expand the subsequent queries for automatically identifying event relevant terms and content.

2. Three Keyword Adaptation Algorithms (KwAAs). By exploring the relationship between event relevance of hashtags and three different features, including the adoption frequency, the traffic pattern and the tweet content similarity between different hashtags, these KwAAs are proposed and integrated with the proposed adaptive crawling model. A thorough evaluation of these against the conventional crawling model over four different type of events is then conducted.

3. A Twitter Event Monitoring solution, called "Detection of Sub-events by Twitter Real-time Monitoring (DSTReaM)". In order to better understand the effects of a broader coverage of event information, this thesis not only explores the impact of data filtering on the sub-event detection, but also compares the detection summary with topical keywords that are identified by the proposed KwAAs.

Other contributions in accordance with the main contributions made by this thesis are:

4. A tweet events corpus that covers 11 different events of various types, including sports competition, music festival, political referendum, nature disaster, crisis protest and etc. Each event in the corpus is retrieved by at least two different methods and with a list of topical hashtags;

5. A novel way for retrieving, aggregating and constructing the vector representation of a single hashtag based on the existing TF-IDF vector calculation, called "Hashtag-based TF-IDF vector".

The above contributions made by this thesis have contributed to the publications listed in Author's Publication:

1.4 Thesis Structure

The remainder of this thesis is organised as follows:

Chapter 2 reviews the relevant context of exiting work on event monitoring, especially with Twitter text streams. First, this chapter gives a general overview of the Topic Detection and Tracking(TDT) project, including the definition of events and an overview of traditional TDT tasks. Then, the characteristics of Twitter and its inherent restrictions in getting event data are introduced. This is followed by a survey of existing event monitoring systems in terms of event tweets retrieval and Twitter event detection. This chapter finishes with a discussion of the existing literatures for bridging the research gaps between the current state-of-art methods versus the problem requirements (to be solved by this research).

Based on the research gaps defined in chapter 2, the thesis then proposes an adaptive microblog crawling model to expand the coverage of the event information, as illustrated in chapter 3. The working mechanism of the proposed adaptive Twitter crawling model is to detect emerging popular event terms and to monitor them to expand the subsequent queries for retrieving highly associated data for the events of interest. Based on the characteristics of live Twitter stream, three Keyword Adaptation Algorithms (KwAA) were designed and integrated to the adaptive Twitter crawling model. With the aim of validating the working efficiency and effectiveness of the KwAAs, this research evaluates the performance of different KwAAs based on the event content they identified from various type of events. The results show that the two (both crawling model and KwAAs) working together can incur at least 20% more event relevant Twitter traffic.

Chapter 4 investigates the usefulness of adaptive crawling for sub-event detection. The Twitter event monitoring solution "Detection of Sub-events by Twitter Real-time Monitoring (DSTReaM)" is proposed and tested with two real-time events. The aim is to

demonstrate that a better event monitoring can be achieved with a broader coverage of event content. In addition, this research also investigates the impact of data filtering on the event-detection results to identify deficiencies of existing algorithm when using the adaptive datasets.

Finally, chapter 5 gives the conclusions for the research work in the whole thesis, and then outlines some of the selective aspects of the research as the recommendation for the future work.

Chapter 2

Event Monitoring by Mining Microblogging Stream

The human desire for getting knowledge about both planned and unplanned events drives the evolution of the modern news media. For a long time, the traditional news media, acts as the principle channel for the general public to get knowledge about events in the world around them. With an increasing number of reports on the worldwide events from diverse news organizations, it can be overwhelming for readers to discover interesting stories from the massive amount of reports. In order to identify specific sets of stories that interest readers, researchers have proposed the Topic Detection and Tracking (TDT) framework for detecting the newsworthy events. However, this TDT framework and its solutions are designed for online newswires that only contain structured and formal news reports.

Since traditional media and online media reports are produced by a smaller number of news professionals compared to the normal public, they can be misleading or biased. User generated event descriptions on online social media, such as Microblogging services can provides more comprehensive information [60]. While some researchers have tried to adapt the existing TDT solutions to an online social media scenario, other researchers have explored new ways to undertake event and sub-event identification. This is because traditional TDT solutions are unable to scale to process the massive amount of streaming

data generated by the online social media services users.

This chapter continues with the background relating to the event detection, including the definition of event in social media (in section 2.1), the fundamental concepts, techniques and evaluation metrics used in conventional TDT framework (in section 2.2), as well as the challenges raised by Web 2.0 applications, i.e. social media services, in the event detection tasks (in section 2.3). Research that aims to improve event monitoring in social media environments are introduced and discussed (in section 2.4). Based on the research questions and a critical analysis of current solutions, this chapter concludes with a summary of the limitations of existing solutions and highlights the motivation of this research work (in section 2.5).

2.1 Events in Social Media

The term "event" is actually very abstract and can be mentioned in various specific domains, such as time series, textual news and social media. This section aims at clarifying the event-related definitions and concepts in the social media environment.

2.1.1 Events and Sub-events

In the scope of Information Retrieval (IR) research, there exists multiple efforts in defining the concept of event. For example, Allan defines an event as "a specific thing which is associated with a specific time and place along with all necessary preconditions and unavoidable consequences" [1], while Yang et al. consider events as "something that are non-trivial and happen at a certain time period" [61]. Given the above definitions, an animal that gives a birth in the wild can be regarded as an event. However, things like this normally don't attract people's attention and can hardly be discussed over social media. Different definitions will lead to different results. Therefore, the formal definition of event and sub-event in this thesis are given as follows:

Definition 1: an **event** under social media environment is something which happens in the real-world at a certain time period and receives constant discussion by social media

users during that time period.

Definition 2: an **event stream** is a set of temporal coherent text pieces which are overlapped in vocabulary. The overlapped vocabulary concerns a common event in social media. An event stream can be represented as a sequence of tuples that includes a timestamp and a set of features, or terms.

Definition 3: a **sub-event** describes the episode of an event by the underlying story or sub-sequent story. The content of sub-events belonging to the same event made up the event stream. As a result, each sub-event can also be represented by tuples which are strongly correlated with each other by the features and coherent by the timestamps.

2.1.2 Event, Topic and Trend

To avoid potential ambiguity when describing event monitoring methods, the definitions and relationship between terms like "topic", "trend" and "event" under the subject of event monitoring are explained in this section.

Some researchers interpret "topic" and "event" in an intuitive way, stating that events are the instances of topics [61]. Another work also recognizes this concept by defining the "topic" as "domain", which abstracts the essence of a set of events belongs to a particular type [62]. Other researchers defines "topic" in a more specific way. A topic is regarded as "a seminal event or activity along with all directly related events and activities" [63]. Namely, a topic is a set of stories which describe the same event. These stories, also known as topically cohesive segment of news, include two or more declarative independent clauses about a single event [3].

In addition to the discussion between "topic" and "event" in the literature, Yang et al. also proposes some other insights about the relation between "trend" and "event" [61]. When they observe the time frequency sequence of a topic stream, they found that the stream always consists of bursts of documents with time gaps. Therefore, they conclude that those gaps indicate that each burst corresponds to an independent event. This observation then leads to the discussion about the concept of "trend". In an earlier work

which reviewed the trend detection methodology for textual data, "trend" is defined as "a topic area that is growing in interest and utility over time" [64]. According to their definition, the objective of trend detection tasks over textual data can be summarised as the identification of topic areas that are previously unseen or are rapidly growing. The rapid growth can be observed by the sharp increases in some features (such as the frequency of terms or the volume of reports) in a text stream [65]. As a result, it becomes very common to use the term "burst" or "peak" to describe the rapid growth of feature in many literatures [65, 66]. As a result, researchers borrow the techniques in anomaly detection [67] or outlier detection [68] since they have the same target: to identify the previously unseen observations that don't follow the regular pattern in a continuous stream.

2.1.3 Event Categorisation

Before monitoring events, the differences between events should be clarified. In general, events can be differ in scale, duration, content and etc. Some research work has been done on categorising events. For example, events can be classified according to their difference on subjects (for example, technology, idiom, sports, political, games, music, celebrity, movies) [69]. Some other work tries to characterise events based on their inherent features. In their classification framework, events are classified as planned or unplanned event [70], trending or non-trending event [71]. According to their definition, an event is considered as planned event only when its title and occurring time are known in advance. Events that have one or more features that are substantially unusual than expected are considered as trading events. The planned event and trending event are not mutually exclusive event type (shown in Figure 2.1). For example, an event should be either planned or unplanned, but can be both planned and trending. Though an event can be characterised using different criteria, the characteristics of different types of events vary significantly. Sometime, it is also hard to classify the event reports even they are about the same type [72].

	Planned	Unplanned
Trending	2012 London Olympic Games	MH17 Crash
Non-Trending	Youtube Geek Week	Traffic Jam

FIGURE 2.1: Examples of Event Categories (reproduced from [70])

2.2 Traditional Event Monitoring: Topic Detection and Tracking (TDT)

The TDT research has long been addressed in the literature since late 1990s. The initial motivation is to provide core technology for news monitoring tools from multiple sources of traditional media (for example, printed media, broadcast media and newswire) to keep users updated about news event developments. As the core research project for providing solutions to event monitoring over traditional media, it lays the foundation for modern event monitoring. In this part, an overview of traditional TDT research is given in section 2.2.1. Their detection tasks and adopted evaluation metrics are described in section 2.2.2 and 2.2.3 respectively.

2.2.1 Overview of TDT

The TDT benchmark evaluation project is initiated and sponsored by Defence Advanced Research Projects Agency (DARPA) of U.S Government [73]. The TDT Pilot Study is explored and conducted by DARPA with additional three institutions, including University of Massachusetts, Carnegie Mellon University and Dragon Systems [3].

With the objective to improve the automatic monitoring of topics from multiple sources of traditional media, a significant massive research efforts have been put into the TDT project. Based on the characteristic of the emergence and development of events in news streams, the TDT project mainly deals with the following five tasks [74]:

- **Story Segmentation:** Identify the boundaries between topics from a topically cohesive continuous stream, and then detect those topics by segment the stream precisely. This task is primarily audio-based.

- **Topic Tracking (TT):** formulate the storyline of a known topic by keeping track of stories similar to a set of example stories
- **Topic Detection (TD):** group the stories that discuss the same topic into single cluster. That is to say, each topic can be represented as a list of stories that are topically correlated.
- **First Story Detection (FSD):** detect if a story is the first instance of a new, unknown topic. The difference from the Topic Detection task is that the output of this system is individual story of each topic.
- **Link Detection:** detect whether or not two stories are related to the same event. Topics can vary differently from each other, so the system needs to adapt itself accordingly. Therefore, the difficulty of this task is to design a detection model requires no prior knowledge.

2.2.2 Tasks of TDT in Event Monitoring

Although these five tasks deal with different problems in TDT research, some of them are closely related and thus are tackled within one solution. For example, the FSD and TT tasks are closely connected with the TD task. These five tasks can be integrated to fit more generalised detection tasks [2].

This integration is based on the observation that some of the real-world topic can be discontinuous. Some topics may become trending again after a period of silence. Taking the MH370 missing plane[1] for example, people mentioned this in Twitter on its one year anniversary, also some details about the missing were revealed discontinuously. Consequently, this characteristics lead to two research questions in the TDT project:

1. How to differentiate stories that belongs to the same topic?
2. How to determine whether the topic emerged previously?

[1]MH370 Missing Plane: `https://en.wikipedia.org/wiki/Malaysia_Airlines_Flight_370`

The appearance of these questions not only leads to the research on topic granularity, but also triggers the shift of research interests from topic to event. As a result, current TDT projects mainly include solutions for two tasks: New Event Detection (NED) and Retrospective Event Detection (RED). In fact, the NED task is equivalent to the FSD task while the RED task is a supplementary. The solutions of these tasks are built upon the solutions of the existing TDT tasks.

New Event Detection (NED) NED task is proposed to solve the first question. This task is very similar to the FSD. Both of them are designed to identify the very first report though the NED system also concerns whether existing report in the system also refers to the same topic.

Retrospective Event Detection (RED) The objective of RED task is to search the report retrospectively to distinguish all the events which refer to the same topic. Namely, it assists the FSD system to review the whole corpus and identify the events that correlated to the topic of interests.

2.2.3 Common Evaluation Metrics

As a research problem under the Information Retrieval subject, the evaluation metrics used in information retrieval can be adopted to assess the TDT solutions. Much of the existing TDT research tends to use one of the popular metrics "Precision and Recall". These two metrics are designed to predict the relevance of a document. Specifically, they measure how precise and complete the retrieved documents are on all the relevant instances. In the TDT frame, they measure how precise and complete the identified events are on all the real-world (realistic) events which appears in the document stream. The definitions of these two measurements can be better elaborated with the confusion matrix, as shown in Table 2.1.

Based on the confusion matrix, the **Precision** (P) is the fraction of retrieved documents that are relevant to user's interests, while the **Recall** (R) is the fraction of relevant

TABLE 2.1: Confusion Matrix for Information Retrieval

	Relevant/Realistic	Irrelevant/Unrealistic
Retrieved/Detected	true positives (tp)	false positives (fp)
Non Retrieved/Not Detected	false negative (fn)	true negative (tn)

documents that are retrieved. As a result, in TDT evaluation, the precision is the fraction of detected events that corresponds to the realistic event, while the recall is the fraction of realistic event that are detected. They can be calculated by equation 2.1 and 2.2 respectively.

$$P = P(relevant|retrieved) = P(realistic|detected) = \frac{tp}{tp + fp} \tag{2.1}$$

$$R = P(retrieved|relevant) = P(detected|realistic) = \frac{tp}{tp + fn} \tag{2.2}$$

Sometimes, the researchers employ the **F-measure** to trade-off both precision and recall by weighted with their harmonic mean with equation 2.3

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R},\ where\ \beta^2 = \frac{1-\alpha}{\alpha} \tag{2.3}$$

However, for modern information retrieval system which got thousands of relevant documents even can't be fully identified, recall is no longer a meaningful metric. In this case, **precision at k documents** ($P@k$) becomes more useful. This metric measures the number of relevant document on the top ranked results thus requires a way to rank the retrieval results.

2.3 Microblogging Text Stream

Microblogging is characterised by the nature of allowing general public to post the short text pieces. Compared with traditional blog posts, a micropost is easier to read and faster to spread, due to its short length [75]. People use microblogging services for not only chatting and communicating, but also for sharing information and reporting news[76]. Consequently, such research tends to analyse posts from Microblogging ser-

vices as they accumulate and distribute event information from the general public. In this thesis, Twitter is used for carrying the research work, as it is one of the most popular microblogging services. With a brief introduction on Twitter properties that are relevant to this research (in section 2.3.1), this section then introduce the event corpus available 2.3.2. Finally, the feasibility of real-time processing with Twitter streaming are discussed (in section 2.3.3).

2.3.1 Twitter as Microblogging Service

As one of the major social media services, Twitter is a popular microblogging site having hundreds of millions of registered users. It is a simple version of blog service and allows users to post short messages (i.e. tweets) up to 140 characters. Apart from the normal web published tweets, users can also access and publish their thoughts on Twitter by the mobile phone that connected to the Internet or Short Message Service (SMS) message.

2.3.1.1 Service Overview: Characteristics

Twitter allows a kind of loose relationship: User X can follow user Y and view Y's contents without requiring approval or a reciprocal connection from user Y. By default, all the posted messages are visible to anyone, but user can set their privacy preferences so that their updates are only available to their friends. These posted messages are displayed as stream on users' main page in a reversed chronological order. Normally, a tweet is represented by a JSON[2] document. This lightweight data format defines the tweet attributes by its properties, e.g. the text, timestamp, URLs, hashtags, user mentions, user information and etc.. The location information will also be presented if applicable. Figure 2.2 shows the structure of a typical tweet in JSON format.

2.3.1.2 Conversational Usage of Twitter Symbols

Twitter is described as "the SMS of the Internet" due to its conversational characteristic. This is supported by its well-known @ mention, RT retweet and # hashtag annotation.

[2]JSON: `http://json.org/`

By placing the "@" symbol before a username, Twitter user can create a mention or reply to the username this symbol linked with. This symbol can be put anywhere in the tweets when the purpose is to mention another Twitter user. However, @ must be put at the beginning of the tweet when it is a response to the mentioned user.

Twitter also allows users to forward (i.e. retweet) other's tweet. A "RT" prefix is used, followed by the user name that creates this message. In fact, this symbol not only disseminate interesting information on Twitter but also increase the influence of the original user.

As the topic indicator on Twitter, the #hashtag annotation allows users to indicate what the message is about when they publish a tweet. By adding a "#" mark before the topic words, users can generate their own topic indicator at any moment. As designed to support grouping similar tweets, the Twitter's user interface automatically associates a hyperlink for each hashtag to allow people to retrieve all tweets with the same hashtag by just one click.

2.3.1.3 Twitter API and Rate Limits

Twitter provides three public APIs to developers and researchers for designing and implementing customised data analysis tools: *Search API*, *Representational State Transfer (REST) API* and *Streaming API*. In the free access manner, it is not possible to retrieve all the Twitter data as rate limit is introduced to restrict the access of API [3].

The Search API and the REST API share similar rate limit. In the older version (V 1.0), an OAuth-enabled application could initiate 350 query requests. While API v1.1's rate limiting model allows for a wider range of requests by 180 calls every time window, i.e. 15 minutes.

Twitter Streaming API is the only accessible interface that offers real-time access to Twitter's public timeline. With the default access level (free of charge), a small proportion of all public tweets , i.e. 1% of whole tweets, in its core database can be retrieved

[3]Twitter rate limits: `https://dev.twitter.com/rest/public/rate-limiting`

by using sample function for each normal OAuth[4] enabled user.

The 1% rate limit is also applied to the filter method of Twitter Streaming API. The filtering function allows the applications to query the core database for extracting all tweets associating with some specific criteria, such as users, keywords, URL link, language, location and etc.[5] However, the full access of retrieval contents is possible only when the retrieved volume is less than 1% of the total Twitter traffic. Otherwise, that 1% will spread out across keywords, only a subset of tweets will be retrieved for each individual keyword. In addition, the number of terms can be tracked is limited to a maximum of 400 keywords for a single query when using Streaming API. However, Twitter provides the fuzzy matching of phrases rather than exact matching. For example, when using "Twitter" as keywords, the engine returns not only its lower case and upper case, but also the tweets which contain it with the quotation, #-hashtags, @-mention and URL format, as shown in Table 2.2 According to the documentation of Twitter, all the

TABLE 2.2: Track Example for Twitter Streaming API Filter Function (reproduced from [78])

Parameter Value	Will match...	Will not match
Twitter	TWITTER, twitter, "Twitter", twitter., #twitter, twitter, `http://twitter.com`	TwitterTracker, #newtwitter
Twitter's	I like Twitter's new design	Someday I'd like to visit Twitter's office
twitter api, twitter streaming	The Twitter API is awesome; The twitter streaming service is fast; Twitter has a streaming API	I'm new to Twitter

rate limits in Twitter are considered on a per-user basis. In other words, if multiple applications belong to the same user account, the rate limits are distributed to each of them.

In terms of the rate limit on tracking tweets, Twitter doesn't release the mechanism

[4] an open standard for authorization, details in `http://oauth.net/2/`

[5] more details about tracking in: `https://dev.twitter.com/streaming/overview/request-parameters`

of how the sampled stream is constructed, and also prohibits third parties from reverse engineering the sampling process. Therefore, some researchers have focused on analysing how the sample stream is constructed. For example, by comparing simultaneous samples from Twitter's Streaming API, Joseph et. al conclude that Twitter sends all connections tracking the same keywords approximately the same result, with over than 96% tweets being the same [79]. Morstatter et. al have analysed what biases are introduced by the Twitter sampling strategy. They discover biases exist in the hashtag distribution and the results from topic detection by comparing a 1% sampled Twitter stream against a random sample of Twitter stream and the entire Twitter stream [80].

2.3.2 Twitter Event Corpus

There exists multiple research efforts on building the comparable event detection corpus [84, 109, 110]. Some of them simply provide unfiltered sample stream, while others labelled the tweets with topic or event tags. For example, Sasa et al. deliver the unfiltered dataset with a list of first stories and on average 112 associated tweets, while Andrew et al. releases the relevance judgments containing more than 150,000 tweets about more than 500 events. However, when considering the number of tweets for each event, it is clear that the event resolution of all these corpuses is limit. Recently, the world's largest event dataset is publicly available by Global Data on Events, Location, and Tone (GDELT)[6]. It provides the event ground truth for researchers to annotate their detection results [39].

2.3.3 Text Stream Mining

Given the real-time nature of the Twitter services, the event tweets are extracted and analysed in a continuous stream manner via the Twitter Stream API. Statistic shows that more than 500 million of tweets are sent every day, with the highest one-second peak record of 143,199 Tweets per second [57]. In this sense, the analysis of Twitter text stream is based on a set of unbounded data without priori known features. Considering

[6]GDELT:`http://gdeltproject.org/`

the huge volume and high velocity of Twitter text stream, it is impractical to store all the data and execute queries that consider all the past data. Processing such text stream in online manner can only be achieved using data stream algorithms [81]. As a result, it is necessary to figure out how text streaming mining related to TDT tasks and its requirements for proposing suitable solutions that work in Twitter.

2.3.3.1 Text Stream Mining and TDT

The input of TDT tasks is a stream of topically coherent materials, normally in text format. To analyse the streaming data, researchers adopt the conventional data stream mining techniques for implementing solutions for TDT tasks. For example, techniques for burst detection [66], clustering [82] of text stream mining is widely adopted for finding coherent or novel stories from a topic stream. Also, in order to reduce the computation complexity and thus improve the analysis efficiency, sketching [52] and hashing [32] which are proposed for data stream mining are employed in TDT solutions. To detect events from Microblogging services, such as Twitter, the principles for designing solutions used in traditional event monitoring problems could be borrowed.

2.3.3.2 Requirement on Streaming Data Mining

In contrast to traditional batch setting, where the training data is available as a whole, analysis on the text stream can only learn from a potentially endless flow of data which arrives in the temporal order. Compared with conventional data mining, additional requirements should be satisfied [83]:

- **Process an example at a time, and inspect it only once.** For rapidly arriving text stream, such as tweets, random access to the data is impossible. New tweets are expected to be processed when they arrived.
- **Use a limited amount of memory.** Although the distributed system can be used to alleviate the processing and storage issue, a typical streaming algorithm should only keep a minimum level of previous data.

- **Work in a limited amount of time.** In order to process the tweet when it arrives in real-time, the algorithm should not be too complicate. Otherwise, extra processing time risks of loss process of income data.

- **Be ready to predict at any point.** An ideal streaming algorithm should be able to produce the best model after any number of tweets it has observed. In practice, the model can be produced periodically to allow some updating time. However, it is desirable to minimise the waiting time.

2.4 TDT for Microblogging: Twitter Event Monitoring (TEM)

Although traditional news websites are still valuable resources for acquiring event information, researchers start to value the user generated contents from newly emerged Microblogging services. The differences between traditional websites (i.e. news portals and blogs) and Microblogging services, such as Twitter, with respect to resource deployment and contents structure make the adoption of Website-oriented methods to Microblogging post quite difficult. As a result, some research attempts are aimed at finding solutions of how to automatically understand, extract and summarise the text input from massive amounts of "social sensor". In order to review the existing work and differentiate the research contributions in this thesis from others, this section first considers a system pipeline related to Twitter Event Monitoring (TEM) (in section 2.4.1). Based on the proposed pipeline, this section then reviews existing TEM systems with respect to event content retrieval (in section 2.4.2) and event detection (in section 2.4.3). Finally, a discussion of existing systems highlighting the key directions for the research reported in this thesis is undertaken (in section 2.4.4).

2.4.1 System Pipeline and Evaluation

The simplest and most straightforward approach to enable TEM is to apply the traditional TDT solution directly to the Twitter stream. However, this is not desired

since the input and output of TDT tasks for traditional event monitoring is in a format which is very different from tweet (in Figure 2.2). While the input and output of traditional TDT solution is structured, meaningful and event-related reports that are produced by experts (journalists or newscasters), the raw Twitter stream can only provide unstructured and informal text. The meaningless babbles [29], advertisements [30], rumours [31] and useful event information co-exists in the Twitterverse. Therefore, pre-processing and post-processing the Twitter stream become two necessary steps in order to realise the same standard of detection results as the TDT solutions produce. Consequently, a generic workflow for TEM can be observed and summarised from the existing research in this area, as shown in Figure 2.3.

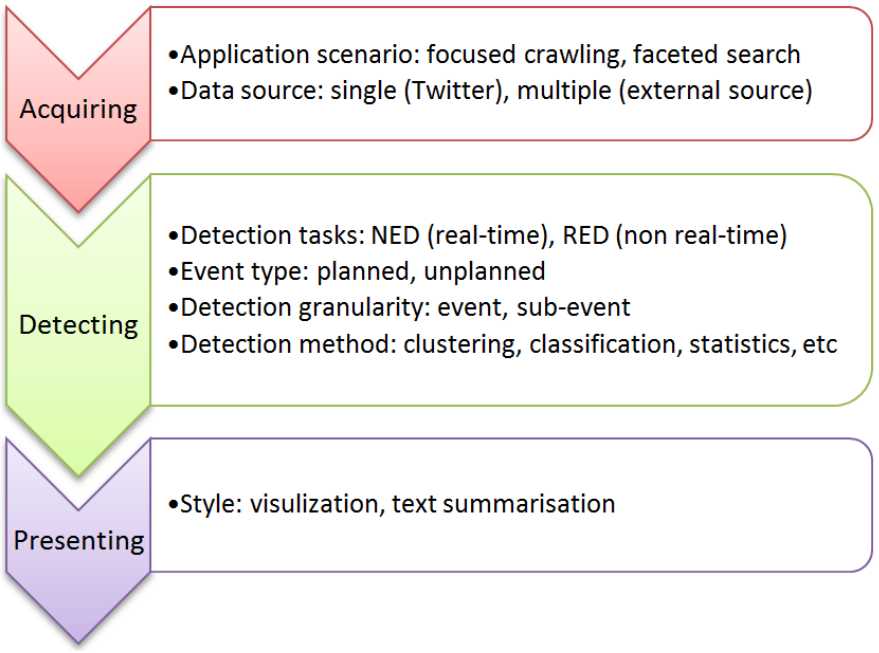


FIGURE 2.2: Event Monitoring Pipeline under Twitter

This pipeline is very similar to the production pipeline of traditional news (in chapter 1). However, rather than producing the news report manually (for example, by news experts), the pipeline of a TEM system emphasises the automatic identification of newsworthy events from the raw and unstructured user input. This research concludes that the production of event information in a TEM system is achieved by three individual procedures: 1) **acquiring** event information from general public; 2) analysing the raw data for **detecting** ongoing events, and 3) synthesizing the detection results for

presenting event summarisation to users, as illustrated by Figure 2.3.

Acquiring event information Since Twitter streams are very dynamic and informal, it is necessary to minimise the amount of noise prior to event detection. As a result, the acquisition process aims at identifying and retrieving a comprehensive set of tweets which are relevant to the events of interests. This additional process of filtering noisy data actually is the main difference between conventional TDT framework and the TEM system. Researchers tried different ways to filter out Twitter noise. Most of them just retrieve the Twitter stream by using pre-defined and constant search criteria, such as keywords, #-hashtags, @-user mentions, URL links and geo-locations. This way of data retrieval is known as focused crawling. However, as explained in section 1.1, the pre-defined keyword strategy risks losing event information and midst event topics. In order to expand the coverage of event tweets, some of the researchers explore the techniques used in faceted search [20, 94], while others rely on additional data from external resources [49, 50]. A more detailed review about this procedure will be discussed in the following section 2.4.2.

Detecting ongoing events After the event content acquisition, all the event tweets are passed on to the next component of the TEM system. In this step, the event tweets are analysed, grouped and extracted for event detection. In fact, developing good event detection algorithm which is applicable to all kind of events is currently a hot topic for research. Based on the detection task, the algorithm can be used in real-time for New Event Detection (NED) [32, 38], or non real-time for Retrospective Event Detection (RED) [48, 52]. The NED algorithms are commonly applicable to the RED task, but the accuracy is much lower than that of RED algorithms. On the other hand, when using the RED algorithms to solve the NED problem, the calculation cost can be enormous. In order to reduce the impact of the different detection tasks, some research scopes their solution to a specific type of event (e.g. planned [49] or unplanned [54] event), or different detection granularity (i.e. whether supporting sub-event detection). By using different detection methods, the outputs of this component can be diverse. Some common output formats are groups of terms [36], clusters of tweets [34], or even time stamps pairs [43].

Although these outputs are based on a different detection method, the main target of this procedure is to detect and extract newsworthy events, abstract by Twitter symbols, from the massive amount of user input. However, existing detection algorithms are all designed for the Twitter stream that is retrieved using fixed search criteria. The survey in section 2.4.3 will review existing solutions and address their limitations.

Presenting event summarisation The final important component for the TEM pipeline is the result presentation. The aim of this procedure is to synthesize and analyse the output from the event detection algorithm for generating either a visual demonstration or a text summarisation. Some existing research presents the raw output of an event detection algorithm, without further processing, by visualising them graphically, such as charts, words clouds, traffic rivers and world maps[106]. Some others present multiple components as a mashup [105]. However, due to the noisy and dynamic nature of tweets, the extracted and abstracted outputs from the "detecting" procedure are normally very hard to interpret. As a result, the essential requirement on presenting the results of event detection is to reduce the result set in order to create a summary which retains the most important event information. Text content is the most favourable medium. This can be either a small group of key phrases or limited number of tweets.

- When describing an event by terms, the top weighted terms are commonly used for event summary, one can weight the terms based on their TF-IDF value [43], distance to the centroid of cluster [42], or auto-correlation of the wavelet signal [33]. Later, researchers tried to improve the readability of summarisation with post-processing. They use a group of terms or segments whose frequency bursts appear correlated to describe the events [36, 103].

- Some researchers directly select the user input tweets for summarisation, since they found that a list of key phrases (e.g. terms, tweet segments) sometimes loses the semantic context. The most descriptive and informative tweets of an event can be select by their distance to the centroid of the cluster centroid [93], the sum of the term weights, such as the k-core number of word co-occurrence graph [47], or the normalised average TF-IDF score [48]. Recently, research on multi-post

summaries concluded that simple frequency based summarisers produce the most promising performance[108]

Although the existing Twitter event corpora offer benchmark for TED system, they are not suitable for mining event with expanding corpus, especially solution for sub-event detection. With different research scenarios and objectives, a great amount of research work exploits their own datasets and evaluation framework for assessing the performance of the event detection algorithm. As a result, there exists no universal evaluation setup for the entire detection algorithm, though their metrics are adopted from the TDT project, or even the IR system (as described in section 2.2.3). The key problem in this step is how to generate the description of event ground truth for comparison. In fact, the main stream media reports [46, 87] and Wikipedia[7] [36, 99] are the main resource pools for this task. Based on the headline or even content of the news reports or the title of the Wikipedia page, the event ground truth is selected and described by a short summary or a list of keywords. Researchers rely on these manually identified standards for measuring the event precision and recall for all the identified events.

2.4.2 Acquiring: Event Tweets Retrieval

Twitter provides multiple APIs for developers and researchers to retrieve the most recent user posts with certain restrictions, as mentioned in section 2.3.1.3. Both the Twitter Search API and Streaming API are the preferable channels in the TEM system, but Streaming API provides better support for real-time applications.

Without any criteria or processing in the acquiring step, some of the early studies crawl random sampled data directly from the Twitter Streaming API [36, 38, 84]. Namely, their datasets are the sampled stream that contains almost every kind of tweets, including breaking news events, advertisements or even people's daily chats.

When the research has more specific requirements, the retrieval is achieved by querying some specific properties, i.e. keywords, timestamps and user. This kind of application

[7]Wikipedia: `https://en.wikipedia.org/wiki/Main_Page`

is known as the focused crawler. Although this way of collecting event tweets loses meaningful event information, it is the most common way for event tweets acquisition. In twitter, crawling a set of online documents, relating to an event of interest can be achieved by tweet properties or Twitter symbols, such as keywords, #-hashtags, @-mentions, URL links and geo-locations.

- **Keywords and Hashtags:** The most common way of crawling is the keywords and hashtags searching. This approach has been widely adopted by research work on event analysis [85, 86, 87]. For example, Starbird and Palen collected information about the 2011 Egyptian uprising by using the terms "*egypt, #egypt, #jan25*" [86], Nichols et al. collected sport related tweets using "*worldcup*" and "*wc2010*" [87]. Recently, Olteanu et al. curate a list of crisis lexicon that can be used to query disaster tweets [88].

- **User Mentions:** On the other hand, the majority of the research work analyse the user behaviour and influence by collecting data on Twitter username [76, 89], some researcher exploits the Twitter user account for event tweets fetching [90, 91]. This kind of approach chooses the users that involved in or related to the event as the initial seed for collection. For example, in order to analysing the effects of Super Bowl 2012 commercials on the preference of car manufactures, the Twitter account of 11 car-related companies that are advertised during the event are selected as initial seeds. [90]. It is similar to the pre-defined keyword crawling approach as the initial seeds are fixed.

- **URL links:** Although URL links can hardly be used for tweets crawling, researchers usually explore URL in tweets when crawling event content outside the Twittersphere. Priyatam et al. used URL links in tweets for domain-specific web content searching [92]. Specifically, their system tries to identify a set of seed domain specific URLs from a Twitter URL graph. With a set of manually generated keywords, their system queries a local Twitter corpus to extract tweets with URL links so to generate the URL graph.

- **Geo-locations:** Geo-location is also a highly preferred property for tweets crawl-

ing [38, 93]. This tweet property tends to be used for tweets selection when the research concentrates on local affairs.

In order to improve the comprehensiveness of the retrieved event information, some research considers this as an application of the faceted search. For instance, Fabian et al. leverage several metrics from Twitter, such as users' profiles, semantics meanings and metadata of tweets, to design the faceted search strategy and to generate new queries for information retrieval [94]. However, rather than to collect event tweets for real world ongoing affairs, the main focus of faceted search is to improve the user experience of interactive searching or to formulate a better ranking strategy in order to select the most informative results [95, 96].

Other researchers attempt to integrate data from additional sources for identify event tweets. For example, Becker et al. examine the use of precision and recall-oriented strategies to automatically identify event features for updating previous queries to retrieve additional event content from diverse social media sites [49]. Their approach is similar to the idea behind the relevance feedback in IR system. Unlike the traditional relevance feedback model which updates the queries by user behaviour or judgments, they rely on event announcements from Eventbrite[8] and other Social Media Sites. This feature is similar to the Pseudo Relevance Feedback (PRF) as the new query is generated based on the results of previous query without an extended interaction. Some researchers use the same idea of PRF but rely on Semantic Web. They associate tweets to a given event using query expansion based on the relationships defined on Semantic Web [50]. The main assumption of these solutions is that prior knowledge about the event is known.

2.4.3 Detecting: Twitter Event Detection

Based on the aspects that are listed in Figure 2.3, this section details the existing Twitter event detection solutions from: the *detection task*, the *event type*, the *detection granularity* and the *detection method*, as illustrated in Figure 2.3.

[8]Eventbrite: `https://www.eventbrite.co.uk/`

2.4.3.1 Detection Tasks

As defined in the previous section 2.2.2, the main tasks that can be applied for Twitter event detection is the NED and RED. Most of the available systems identify event in an offline retrospective manner, as their objective is simply to cluster tweets which talk about different events [33, 36, 38]. Later on, some researchers try to enable the online NED using distributed computing [99, 100], while others tries to reduce the number of comparisons which need to be done during the detection process. For example, Petrovic et al. presents a real-time Twitter event detection algorithm based on Locality Sensitive Hashing[9] [28]. By hashing the similar tweet with same hashcode, the algorithm can detect new events from Twitter stream in real-time. Similarly, Becker et al. reduce the number of comparison by comparing tweets only with the centroid of existing event clusters. [34, 93].

2.4.3.2 Event Types

As described in section 2.1.3, events can be categorised according to their characteristics. From a generic view, it is possible to simply classify event into planned and unplanned. Information about the planned event is easier to obtain in advance to the event, and thus becomes the main research subjects for the majority of existing solutions. For example, researchers mined the Twitter stream for analysing sport events [10, 26, 97], broadcasting political events [25, 51, 52], festival events [49], transportation event [98] and etc. On the other hand, some researchers focus on tailoring solutions for unplanned events [42, 85, 94] that are disasters or emergent incident, such as fire, earthquake and tsunami. The main target of these work is to provide people with better situation awareness during the aftermath of emergencies. As different emergencies are distinct in terms of their characteristics, they tend to be analysed separately for Twitter event detection.

[9] This technique hashes input item to reduce the dimensionality of high-dimensional data by maximize the probability of "collision" for similar items.

2.4.3.3 Detection Granularity

Most of the existing solutions to the Twitter event detection are designed for grouping tweets in the dynamic Twitter stream into different event clusters [33, 36, 38]. In other words, these research investigates the entire Twitter stream and detected multiple events that are independent with each other. The detection of sub-event is also explored in recent year [42, 44, 47, 48, 51, 52]. In these research scenario, the event stream is separated according to the episode of the event. Therefore, the detection results reveal how the event is evolved with different sub-events.

2.4.3.4 Detection Methods

For both the traditional TDT tasks and the TEM system, the techniques used in the outlier detection are widely adopted. The main target is the identification of event content (tweets or reports) which is not similar with other event content in the income stream.

The classification techniques are explored in the Twitter event detection. For instance, Popescu et al. identify the controversial events from Twitter by training the Decision Tree using the manually annotated training set. This set is composed of controversial event, non-controversial event and non-event examples [101]. Later, Sakaki et al. estimate the location and trajectory of disaster events by devising a classifier based on the temporal and spatial features of tweets [85]. Chierichetti et al. build a classifier with the volume of tweets and retweet about the broadcasting event for identify important sub-events [45]. This research is based on the fact that users become less social (less volume of retweet) when the event just happen, but quickly back to socialising afterwards when seeing the broadcasting of event. As the supervised machine learning approaches, these solutions require very detailed priori knowledge on the events to be detected. It is compulsory to provide both positive and negative instance for training an unbiased classifier on the events.

The online incremental cluster analysis [102] is another type of machine learning tech-

niques which is widely adopted in the Twitter event detection. In this case, an stream of text content is grouped into different clusters based on the similarity between one or multiple features. The output of this kind of method is a set of clusters that consist of multiple tweets. When applying the algorithm over the entire Twitter stream, a cluster can be an event cluster or non-event cluster [32, 34]. When the input is the event stream, the output clusters are considered as different sub-events [44, 52]. The common research interests in this approach are to discover useful features and similarity measurements in order to achieve better event detection result. Some commonly used properties for tweets clustering include co-occurrence of terms [33], frequency of terms [20] and tweet metadata (e.g. time and location) [42, 44].

Statistical methods such as burst detection are the most popular methods for the detection of Twitter events. The idea is that an emerging news event can derive people's sudden interest on posting and forwarding tweets about it, and therefore can be associated with the burst of some features [103]. As a result, if the frequency of a feature apparently deviates from an expected value, the algorithm will report an event and describes it with that burst features. For example, Earle et al. identify possible earthquakes by predicting the count frequency with a short-term-average and long-term-average algorithm to identify possible earthquakes [104]. A similar idea was proposed to detect sport events [10]. These solutions detect the events by applying a adaptive window that determines the duration of event based on the relative tweet frequency. Twitinfo system detect event with similar technique but with a more theoretical model. Instead of manually defining how to calculate the average, their work borrows the idea of exponential weighted moving average from TCP's congestion control mechanism for smoothing the frequency count to enable a better event detection [43]. Apart from these detection algorithms, researchers also use different mathematical model to describe the distribution of tweets when a major event occur. Some work considers the probability of observing n features in a time window as a binomial distribution [36, 41]. Exponential distribution is also employed to model the volume of tweets when an event occur [85]. In a recent work, the authors model the tweet stream as a mixture of multiple inhomogeneous Poisson process [38]. Although some methods integrated the burst detection with machine

learning techniques for improving the performance, the assumption on the event remain to be the same [36, 47, 52].

2.4.4 Discussion

Based on the critical analysis given above, it is observed that existing TEM solutions actually are designed for different scenarios. Some research is interested in finding the first story and in tracking the follow-on post within the entire Twittersephere [33, 34, 35, 36, 37, 38, 84]. As a result, they collect the sample stream from Twitter Streaming API to represent the state of the entire Twitter. In addition, their event detection algorithm focus on identify event clusters that talk about events that are significantly different from each other. The granularity of detection is not considered since the vocabulary of different events are normally distinct. However, existing research has shown that Twitter didn't provide quicker information about a newsworthy event [27]. Instead, the advantages of Twitter lies in its a broader coverage of event information intertwined with additional viewpoints [39] and its capability in revealing wider aspects about the evolution of events [40]. Therefore, the FSD is not the interest of this research since an opportunity exists in mining the diversity and the evolution of a newsworthy event through analysing users' input tweets.

In order to take advantage of the more comprehensive Twitter event information, it is necessary to explore methods that expand the coverage of the event information. However, based on the review above, it is clear that almost all the existing research uses pre-defined constant keywords as the retrieval criteria. Namely, these researchers only considered the state of the Twitterverse at a particular point in time. They simple ignore the high probability of vocabulary variations as the event evolves. This research problem is commonly considered as a faceted search problem, i.e. allowing users to explore a collection of information by applying multiple filters, either involves user input or emphasises the accuracy on top ranked items. Although additional metrics [94] and external resources [49] help to improve the accuracy, the issue remains in the nature of interactive process. A fully automatic mechanism that can expand the coverage of event

tweets but that requires a limit amount of processing power need to be developed.

Discovering the underlying sub-events is another important procedure in the TEM system, and it became a research hotspot these years. Initially, research examines the feasibility of using an existing solution to distinguish the sub-events [42, 44, 45]. However, these solutions are tailored for RED task. The classification-based event detection requires a training stage while solution based on clustering techniques concludes the result by analysing large amounts of metrics. Therefore, researchers think of modifying these solutions to fulfil the real-time requirement. Unfortunately, these classification-based algorithms can only be used to discover sub-events for a specific topic. The assumption made by these classification-based solutions are the prior knowledge about event is available. However, events of the same type can have very different characteristics (for example, the vocabulary used in football event and basketball event are different). Moreover, the amount of event types in Twitter stream is hard to define. These factors make it infeasible to train a classifier which is capable to deal with all the events on Twitter. On the other hand, the modification of the unsupervised clustering-based algorithm is achievable. It can automatically group event tweets without prior knowledge. Researchers try to find different events with single pass clustering algorithm and use only tweet content [32, 34, 41]. However, when applying it in the sub-event detection tasks, only very fragmental clusters are generated: most of the clusters are very tiny and the big clusters always maintain a set of near duplicated tweets [52]. Accordingly, statistical-based outlier detection algorithms seems to be the most suitable solution. It works for both planned and unplanned event detection and require no prior knowledge on the events. However, the statistical features of event streams can be hard to model and thus the additional post-processing is required [38, 52]. When the input Twitter stream varies over time, the original model need to be updated accordingly. Moreover, most of these sub-event detection algorithms are tailored to deal multiple instances of the same kind of events [44, 47, 48, 51]. Even the detection algorithm is examined with multiple events, the event under investigation tends to be the same type (either planned event [52] or unplanned event [42]).Therefore, a research gap exists in proposing better event and sub-event monitoring solutions that detect additional newsworthy topics in

real-time by taking advantage of the diverse event tweets.

2.5 Summary

This research focuses on the analysis of real-word events and sub-events, which are trending over social media and continuously receive discussions from general public. Both planned events and unplanned events are considered in this thesis.

As discussed in section 1.2, event detection with a finer granularity is required since Twitter is outstanding in providing a wider coverage of people's opinion. There exist research efforts that have been made to detect sub-events on Twitterverse. However, they all ignore the impact of the evolution of the event on the detection algorithm. In addition, these existing solutions are tailored for specific types of events and are either incapable of running in real-time or require a priori knowledge about the event.

In order to obtain a comprehensive set of event knowledge about all types of event (i.e. both planned and unplanned events) with the underlying and subsequent stories and solve the problem in a real-time resolving manner, a TEM solution that takes advantage of expanded coverage of user inputs is required. This solution is desired to discover and extract sub-events in an efficient real-time manner and requires no prior knowledge.

Chapter 3

Real-time Event Content Identification via Adaptive Microblog Crawling

The widespread use of Microblogging services, such as Twitter, makes them valuable resources to correlate people's personal opinions about real-world news events. Researchers have capitalized on such resources for monitoring real-world news events. In order to identify and analyse events among the entire Twittersphere in real-time, gathering a comprehensive dataset describing the event in a streaming manner is essential. However, current Twitter event monitoring approaches tend to analyse events based upon partial and static datasets which are retrieved by a set of pre-defined keywords. Although, some researchers try to improve the quality of tweets retrieval by synthesizing Twitter data with multiple external resources (such as Wikipedia, Eventbrite and DBpedia), they either rely on additional processing power or priori knowledge on events. The requirements of these solutions make it difficult to apply them on the real-time event monitoring and analysis (as described in Chapter 2).

This chapter deals with the challenges raised by identifying event content in the real-time scenario. This chapter begins with a preliminary exploration of tweets from the 2012 London Olympic Games. It demonstrates that the retrieval of Twitter posts by the

static pre-defined keyword approach risks losing valuable information relating to event (in section 3.1). To overcome this limitation, an adaptive Microblog crawling model (referred to as "adaptive crawling" or "adaptive crawling model" for simplicity) is proposed to extend the conventional baseline crawling model (in section 3.2). The proposed adaptive crawling model can detect emerging popular event topics using hashtags, and monitor them to retrieve greater amounts of highly associated data for the events of interest. Based on the characteristics of live Twitter stream, several Keyword Adaptation Algorithms (KwAA) are designed and integrated to the adaptive crawling model (as shown in section 3.3). To investigate the performance of the proposed adaptive crawling model and the KwAAs, this chapter first addresses the methodology for evaluation (in section 3.4). After that, the configuration of parameters for KwAAs is introduced (in section 3.5). This chapter then evaluate the adaptive crawling model from two aspects: the performance of different KwAAs and the performance across different types of events (in section 3.6). Finally, the overall summarisation on the performance of the adaptive crawling and the characteristics of the KwAAs is listed (in section 3.7).

3.1 Event Content Identification: Solutions and Challenges

In microblogs, people share their observation of events through online social media services. Consequently, Twitter, one of the most representative online microblogging services, becomes the resource pool for researchers to monitoring the real-word news events. Recent research has examined the use of such service to get knowledge about ongoing affairs [49, 94, 105], or even to dig out hints of upcoming events [85, 111].

In order to identify and analyse real world events among the entire Twittersphere, a comprehensive dataset describing the event is essential. As shown in previous section 2.4.2, the majority of collection techniques collect tweets from the live Twitter stream by matching a few search keywords or hashtags. However, the set of predefined keywords is subjective and can easily lead to incomplete and bias dataset [112]. Sometimes, people will communicate their observation and perception about events, even without explicitly mentioning the title of the event [87]. Moreover, even given expert knowledge,

keywords and specialised hashtags often arise in the midst of such events. For example, Figure 3.1 shows two tweets relating to the same football match during 2012 London Olympics Games. It is straightforward to determine that the first one is related to the

> Goal! Aaron Ramsey. Penalty. GB 1-1 Korea. **#football #olympic**
>
> And just like that **#FIFA** awards **#GBR** a penalty. **#GBRvKOR**

FIGURE 3.1: Tweets about Football Competition during 2012 Olympic Games: A match between Britain and South Korea on 2010-08-04

2012 Olympics football event, whereas the second one, which refers to the same event, is much harder to distinguish. Unlike the first tweet, which contains term "*olympic*" and "*football*" explicitly, the second tweet is composed with hashtags that have emerged during the Olympic event (i.e. *#GBRvKOR* and *#GBR*). In fact, tweets that similar to the second tweets is easily missed with the conventional pre-defined keyword collection.

Since lot of tweets are written in the same way as the second tweets in Figure 3.1, the pre-defined keyword strategy will result in the loss of event relevant information, as illustrated in Figure 3.2. This figure is plotted with the datasets retrieved during 2012 London Olympic Games. The red dashed line represents the volume of tweets mentioned "*olympic*", while the red dashed line represents the volume of tweets mentioned "*olympic*" or "*#teamgb*". It is clear that both lines burst around the same moments, but the volume varies significantly. As marked in the orange oval, the volume for tweets that contain either "*olympic*" or "*#teamgb*" is twice as that for tweets only contain "*olympic*", which is the closet difference across the Olympic Game period. If "*#olympic*" is the only search term in the pre-defined keyword set, tweets that only contain "*#teamgb*" will be lost even if they are relevant to the event of interests. Namely, a larger amount of event information can be fetched if keyword "*#teamgb*" is introduced. This issue is even more severe when using Microblogs for unplanned event (e.g. emergencies or disasters). This is caused by the nature of unplanned events: the evolution of them is unpredictable, which makes it even harder to pre-define the keywords.

Moreover, Twitter API rate limits greatly complicate the collection process (as men-

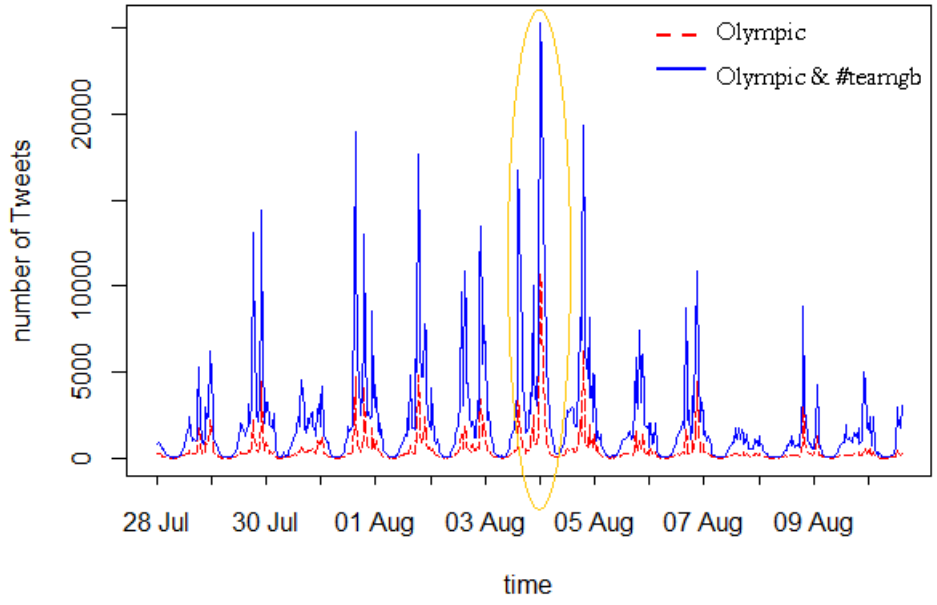


FIGURE 3.2: Comparison of Tweets Volume that Crawled by Different Keywords *Olympic* (lower, red dashed line) versus *Olympic* and *#teamgb*(higher, blue solid line) during the 2012 London Olympic Games

tioned in section 2.3.1.3). The amount of data that can be accessed free of charge is severely restricted. When retrieving live tweets, the rate limits for Streaming API are applied. According to the official documentation, only up to 1% of the total tweets can be fetched. The rate limits not only introduce difficulties on live tweets retrieval, but also make historical crawling hard. As the number of requests within a Twitter time window is limited, getting event tweets afterwards can take long time (usually 1800 tweets per 15 minutes). Moreover, tweets published one week ago are not accessible from the search API. It is only possible to retrieve them directly from individual users' timelines, which is unrealistic in time critical event scenarios. In addition to the challenges in section 1.3, these rate limits restrict the efficiency and effectiveness of the event content retrieval process, and therefore bring impact to the quality of event analysis. As a result, a pertinent and fundamental problem in event detection is how to expand the coverage of relevant information given the rate limits and restrictions, in a real-time, and efficient manner.

3.2 Twitter Crawling Model

A Twitter crawler is a program that collects tweets or users' information through the Twitter API by matching a set of search criteria. Although Twitter provides multiple parameters to track with, keyword tracking is the most commonly used approach in real-world event detection scenarios (as discussion in section 2.4.2). In this section, a novel adaptive crawling model will be introduced. This adaptive crawling model is initialised using simple keyword crawling (baseline crawling model) but is equipped with a keyword adaptation algorithm running in real time. Namely, this research focuses on the keyword-based crawling, where every matching tweet will contain at least one of the defined search keywords.

3.2.1 Baseline Crawling

The baseline crawling model defines and uses a constant keyword set. In this model, a keyword set is used for focused crawling of a particular event. The keywords are manually defined according to the event of interest and remain unchanged for the entire collection period. The system flow of this crawling model is illustrated in Figure 3.3.

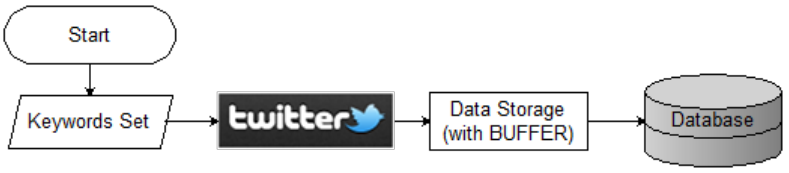


FIGURE 3.3: Components and System Flow of Baseline Twitter Crawling Model

By requesting the Twitter Streaming API with the pre-defined keywords, the qualified tweets (which contains any of the keywords) will be returned as a real-time stream. These tweets are stored in a database system. When evaluating the proposed adaptive crawling model, datasets collected by this model are used as a benchmark since this crawling approach is used by most of the existing research, especially for the real-time analysis.

3.2.2 Adaptive Crawling

The system structure of the adaptive crawling model is similar to the baseline crawling model for the Data Collection and Data Storage Components. The difference is the additional *Keyword Adaptation* component, as illustrated by Figure 3.4. This component

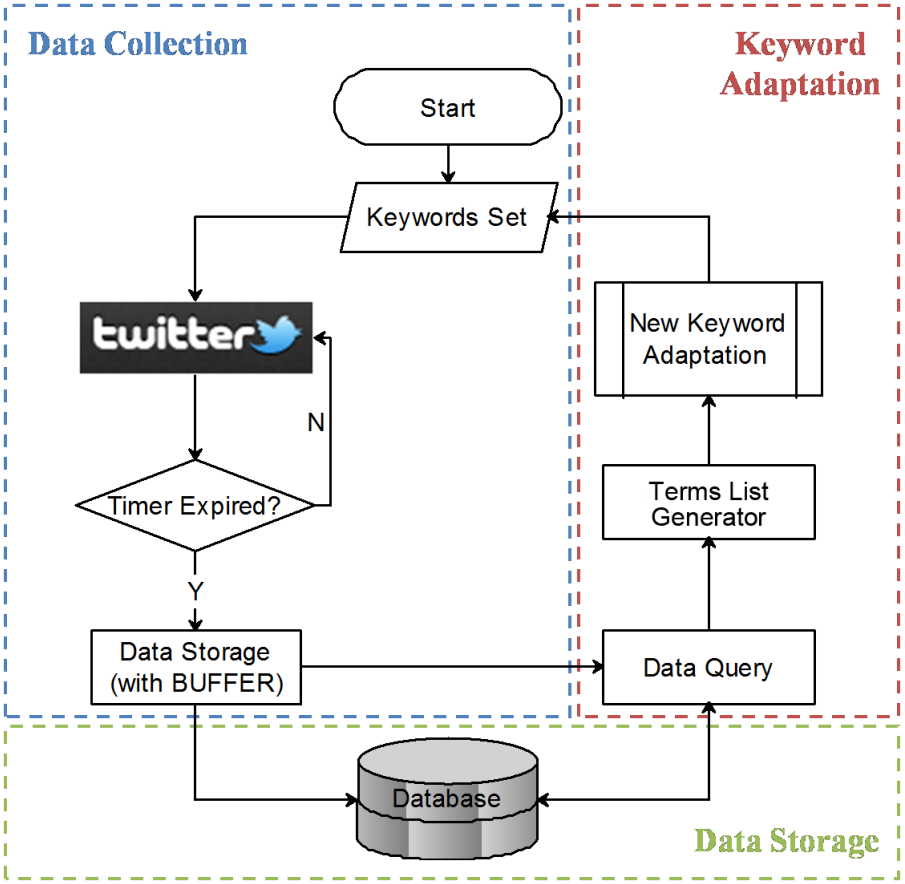


FIGURE 3.4: Components and System Flow of the Adaptive Crawling Model

is in charge of adapting the subsequent search query by including new terms identified in the current keyword adaptation iteration.

In this crawling model, the data collection process is triggered by using the same set of predefined keywords (initial seeds) as the baseline crawler. However, instead of collecting consistently with the initial seeds, the keyword adaptation component enables the identification of popular event-related topic terms as additional keywords (this function is achieved by using the Keyword Adaptation Algorithms (KwAAs) that are detailed in the following section 3.3). Specifically, at the end of each time frame, the data query module retrieves all the content in the last time frame. Then, this data is processed by

the term list generator for a ranked term table. When the list is passed to the new keyword adaptation module, the KwAAs will identify event topic terms and generate a new set of keywords for the event of interests, based on the input term-frequency statistics and the term similarity to the initial seeds. Finally, a query that encodes all the terms in the new keyword set is sent to the Twitter API. At the same time, the timer for the next time frame is reset.

However, the problem for the adaptive crawling is to identify "good" terms[1] that enable more event relevant content to be retrieved. In order to run the adaptive crawler in real-time manner, the process of identifying candidate terms and formulating new queries need to be efficient. According to the review of existing research (as discussed in section 2.4.2), three Twitter's symbols are widely used as the search criteria, which are URL link, user mention and hashtag. This research excludes the use of the URL link since Twitter doesn't support exact match of shorten URLs. It also excludes using user mentions for event content retrieval as extensive user look up is limited by the real-time running requirements of adaptation appropriate. Consequently, this research uses hashtags as the new keywords for the adaptive crawling. This choice is further supported by existing research that has demonstrated that hashtags can link the event with relevant topics when people describe observations and express opinions [113].

3.3 Keyword Adaptation Algorithm (KwAA)

To enable Keyword Adaptation, mechanisms that can select hashtags for collecting event relevant content need to be designed. As a result, this research proposes to solve the selection problem with Keyword Adaptation Algorithms (KwAAs). In the initial attempt, this research applies the simple idea of selecting new keywords based on hashtag frequency, as described in Term Frequency based approach (TF-KwAA). The basic assumption is hashtags that appear more frequently in tweets with initial keywords are related to the event. However, as shown in the evaluation of section 3.4.3, this approach introduces extensive amounts of noise. Moreover, due to the restrictions from Twitter

[1] "Good" terms are those that lead to the collection of event relevant tweets, rather than those that introduce irrelevant tweets.

on the sample rate, only a limited amount of tweets can be retrieved. Among the limited amount of tweets, the noise introduced by TF-KwAA quickly (usually three to four iterations for breaking news events) occupies the space, and results in less event relevant tweets. In fact, the volume of the event-related tweets retrieved by TF-KwAA is far less than the volume collected by simply using traditional keyword crawler. In order to balance the efficiency and performance of crawling content under Twitter API restrictions, two additional algorithms are proposed. Based on the proposed TF-KwAA, the design of the new KwAAs considers two different characteristics of Twitter hashtags:

1) **Traffic Pattern of hashtags (TP-KwAA)** - this approach is based on the assumption that new keywords should have similar frequency count distributions as the initial keywords. 2) **Content Similarity of tweets that represent the hashtags (CS-KwAA)** - this approach is based on the assumption that new keywords should share common vocabularies as the initial keywords in terms of the tweets that represent them. In this section, the full details about all three aforementioned KwAAs are described.

3.3.1 Term Frequency based Approach (TF-KwAA)

TF-KwAA first identifies all the hashtags that co-occurs with the collection of initial keywords (or initial seeds, represented by $H_{seed} = \{h_1, h_2, ...\}$) in the n_{th} time frame t_n, represented as $H_{all}(t_n) = \{h_1, h_2, ..., h_k, ...\}$, where $h_k(k = 1, 2,)$. In this thesis, the term "*keywords*" only refers to the hashtags that are used in the search query for tweets retrieval. The keywords set, sent back to Twitter API in Figure 3.4, in the same time frame, is a subset of $H_{all}(t_n)$ and is represented as $H(t_n) = \{h_1, h_2, ...\}$. $H(t_n)$ satisfies specific criteria, high frequency of co-occurrence with initial keywords. Apart from the hashtag lists, the algorithm also keeps two hashtags frequency lists; $F_{all}(t_n)$ and $F(t_n)$. $F_{all}(t_n) = \{f(h_1, t_n), f(h_2, t_n), ...\}$ is the individual frequencies at all observed hashtags at the end of n_{th} time frame t_n. The frequency list $F(t_n)$, as a subset of $F_{all}(t_n)$, is used to record the frequency of the keywords. The hashtag list and the frequency list have a one-to-one correspondence, i.e. the frequency count of a hashtag h_k at n_{th} time frame is $f(h_k, t_n)$, so does the $F(t_n)$ to $H(t_n)$. The Frequency List Update is defined in

Algorithm 1.

Algorithm 1 Frequency List Update

Require: $H_{all}(t_n)$, $F_{all}(t_n)$

for $\forall h_{in}$ in the incoming tweets **do**
  if $\exists h_k = h_{in} : h_k \in H_{all}(t_n)$ **then**
    $f(h_k, t_n) = f(h_k, t_n) + 1$;
  else
    add h_k to $H_{all}(t_n)$;
    $f(h_k, t_n) = 1$
    add $f(h_k, t_n)$ to $F_{all}(t_n)$;
  end if
end for

When a hashtag h_k appears, the Algorithm 1 is executed to check whether the hashtag already exists in the hashtag list $H_{all}(t_n)$ for the n_{th} time frame. If this hashtag has already emerged, its corresponding frequency $f(h_k, t_n)$ is incremented by 1. Otherwise, both the hashtag list $H_{all}(t_n)$ and the frequency list $F_{all}(t_n)$ are updated to include this new hashtag h_k with a frequency $f(h_k, t_n) = 1$.

To enable an efficient keyword adaptation, a minimum frequency (f_{min}), as a threshold for being a keyword, and an array of blacklist hashtags (H_{black}) are also used in this TF-KwAA (this will be explained later this section). The pseudo code in Algorithm 2 explains the details of this KwAA.

Algorithm 2 Term Frequency based Keyword Adaptation Algorithm (TF-KwAA)

Require: $H_{all}(t_n)$ and $F_{all}(t_n)$ from Algorithm 1

for $\forall h_k \in H_{all}(t_n)$ **do**
  if $\exists h_k : f(h_k, t_n) < f_{min}$ **or** $h_k \in H_{blacklist}$ **or**
  $\{h_k \in H_(t_{n-1})$ **and** $f(h_k, t_n), ..., f(h_k, t_{n-n'}) = 0\}$ **then**
    remove h_k from $H_{all}(t_n)$;
    remove $f(h_k, t_n)\}$ from $F_{all}(t_n)$
  else if $f(h_k, t_n) \in top\ N[F_{all}(t_n)]$ **then**
    add h_k to $H(t_n)$;
    add $f(h_k, t_n)$ to $F(t_n)$
  end if
end for

This algorithm keeps at most N keywords when query Twitter Streaming API. By default, the value is set to be $N = 400$, as it is the maximum number of terms can be used to filter the Twitter Streaming API[2]. When the timer expires (at the end of each

[2]This track limit is defined in Twitter API version 1.0, and has been adopted in the current version

time frame), the hashtags in the hashtags list are sorted according to their frequency. Top ones will be added to the keyword set, while those with low frequency are ignored. This is because that the number of hashtags that co-occurred with the initial keywords can be huge. A random sample of 10 independent time slots from London Olympic dataset shows that, on average, 355 new hashtags emerge in every minute. In order to reduce the number of potential keywords, this research only keeps keywords with a frequency count higher than the median hashtag in the initial filtering step. This preliminary filtering strategy is used to reduce the number of keyword comparisons needed. The 50% threshold value was not investigated in further detail, as in practice it does not impact the results as this bottom 50% always contained hashtags which were also did not adhere to our minimum frequency restriction (described below).

To avoid the overwhelming of non-related keywords in the new keyword set, the following three noise reduction steps are employed in the TF-KwAA:

1. **Minimum frequency**

   This threshold, f_{min}, helps to filter out the unusual and non-related hashtags, especially when the crawler first starts. The introduction of low frequency hashtags will significantly increase the calculation cost, both in space and time, and are very unlikely to introduce useful amount of event-tweets. As a result, f_{min} is empirically set to be one per minute.

2. **Rare keywords discarding mechanism**

   Some newly identified keywords are popular in a specific period of time, but fade away quickly after that. Since the number of keyword is limited to N, a lot of space will be wasted if the algorithm keeps track on these keywords. By discarding the long-term-low-frequency items, the crawler can improve the utility of N keywords. This mechanism functions as follows: any hashtag h_k whose frequency is lower than x for a long period $(f(h_k, t_n), f(h_k, t_{n-1}), ..., f(h_k, t_{n-n'}))$ will be removed from the keywords set.

3. **Modifiable keyword blacklist.**

1.1, see the announcement in `https://blog.twitter.com/2013/api-v1-is-retired`

The introduction of the keyword blacklist allows noisy keyword to be manually filtered. The blacklist is empty when the crawler is started. Users can identify and add non-related words to the blacklist during the collection period. The algorithm will check this list every time when it identifies new search terms so it can discard the words that are in the blacklist. For the experiments in this paper, the blacklist words are either the abbreviation of news channels (e.g. #BBC for British Broadcasting Corporation, #CNN for Cable News Network and etc.) or hashtags used by follow up and follow back activities (e.g. #teamfollow and #followback).

After all the above steps, the number of keywords is expected to be lower than the $N = 400$ limit. If the number of keywords is higher than 400, the TF-KwAA only sends the top 400 keywords with highest frequency back to the Twitter API.

3.3.2 Traffic Pattern based Approach (TP-KwAA)

According to the evaluation results that are presented in 3.6.1, initial attempts show that extra event content is identified when using TF-KwAA. However, the dataset collected through TF-KwAA also contains a large amount of noisy tweets (sometimes is even worse than the stream retrieved by the sample function of Twitter Streaming API). Moreover, the longer the crawler runs, the larger the proportion of noisy tweets. The noise, namely, event irrelevant tweets, eventually overwhelm the event relevant content, which results in a chaotic and meaningless dataset. This issue is caused by the fact that the algorithm relies on the collected content: a clean keyword set will helps the KwAA adapts correctly, while a polluted keyword set confuses the KwAA with noisy hashtags (been wrongly considered as event relevant keywords).

As a result, the problem is how to modify the TF-KwAA so the adaptive crawler collects a greater amount of event-associated data without significantly increasing the dataset noise. In order to reduce the impact of noisy information on the adaptive dataset, the traffic pattern of hashtags, i.e. frequency count distribution of the hashtags, is exploited to identify new search terms. The basic assumption of this KwAA is that the frequency

trends of any event-related hashtags should be similar to that of the initial keywords. In other words, the frequency distribution of a new hashtag should be positively correlated to that of initial keywords. The higher the correlation is, the more similar the two terms are.

The refined version, TP-KwAA, first automatically gets the hashtags list $H(t_n)$ as generated by TF-KwAA. The list is then passed to an extended part of the keyword adaptation algorithm for assessing the elements' relevance to the event. Although the ideal situation is to pass the hashtags list $H_{all}(t_n)$ to the extended part, this research only chooses the subset $H(t_n)$ to avoid the frequent queries to Twitter Streaming API (that are restricted by Twitter rate limits). To measure the relevance, the *correlation coefficient* exploited. In order to calculate the correlation between two hashtags, the original time frame is subdivided into m time slots (as illustrated in Figure 3.5).

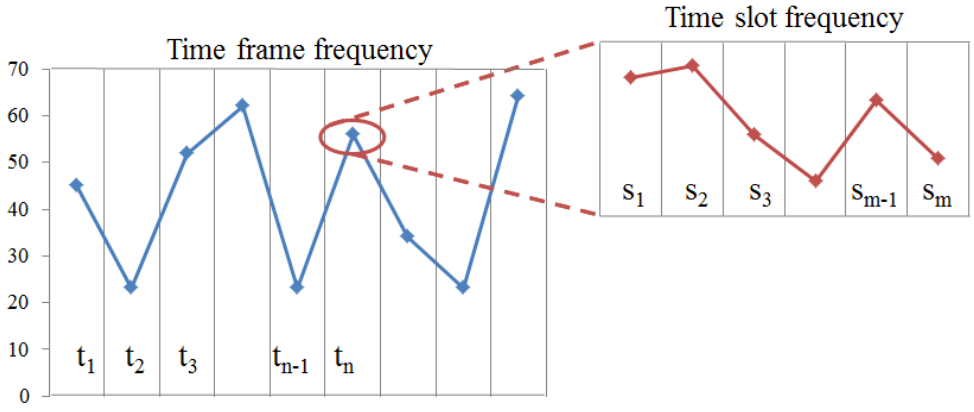


FIGURE 3.5: Time frames and Time slots for Hashtag Frequency

As defined previously, the total frequency count of hashtag h_k at t_n is represented by $f(h_k, t_n)$. Therefore, the frequency count of hashtag h_k for all the slots at t_n can be represented with $F(h_k, t_n) = \{f(h_k, t_n, s_1), f(h_k, t_n, s_2), ..., f(h_k, t_n, s_m)\}$. Instead of using $H(t_n)$ as the input for querying tweets in the next time frame, $H_{fin}(t_n)$, a subset of $H(t_n)$ is used to represent the keyword set. The pseudo code is updated as the Algorithm 3.

The relationship between initial keywords H_{seed} and the keyword set at the beginning of each time frame, as well as the correlation measurements *cor* are defined based on

Algorithm 3 Traffic Pattern based Keyword Adaptation Algorithm (TP-KwAA)

Require: $H_{seed}, H_{fin}(t_n) = \varnothing, H(t_n)$
1: **Execute Algorithm 2**
2: **for** $\forall h_x \in H(t_n)$ **do**
3: **for** $\forall h_y \in \{H_{seed} \cup H_{fin}(t_n)\}$ **do**
4: **if** $h_y \in H_{BL}$ **and** $cor(F(h_x, t_n), F(h_y, t_n)) > Thres_1$ **then**
5: **if** $h_x \notin \{H_{seed} \cup H_{fin}(t_n)\}$ **then**
6: add h_x to $H_{fin}(t_n)$
7: **end if**
8: **else if** $h_y \notin H_{BL}$ **and** $cor(F(h_x, t_n), F(h_y, t_n)) > Thres_2$ **then**
9: **if** $h_x \notin \{H_{seed} \cup H_{fin}(t_n)\}$ **then**
10: add h_x to $H_{fin}(t_n)$
11: **end if**
12: **end if**
13: **end for**
14: **end for**

the following assumptions:

Assumption 1 the initial keywords used for both baseline crawler and adaptive crawler are the most representative words that describe the event of interest.

Assumption 2 keywords for an event during one particular or several sequential time frames are likely to exhibit similar traffic patterns.

Assumption 2.1 the frequency count of two event-related hashtags should positively correlate with each other. Namely, when keyword A appears more frequently, the frequency of keyword B will also increase, and vice versa.

The initial keywords used by the baseline crawler and adaptive crawler with TF-KwAA are also selected as initial keys in TP-KwAA. To measure the correlation between the traffic patterns of hashtags, this research tests the selection of potential keywords with three correlation coefficient measurements, i.e. Pearson's r, Kendall's τ and Spearman's ρ. Through a series of experiments (more details in section 3.5), results show that r and ρ achieve similar performance, and both better than τ. Since the Pearson's r gives slightly better results, this research chose the Pearson correlation coefficient to measure the similarity between keywords. The range of Pearson correlation is between +1 and -1 inclusive, where 1 represents a positive correlation, 0 represents no correlation, and

-1 represents negative correlation. The formula is defined by the equation 3.1

$$cor(h_x, h_y) = \frac{\sum_{i=1}^{m}[f(h_x, t_n, s_i) - \overline{F(h_x, t_n)}] \cdot [f(h_y, t_n, s_i) - \overline{F(h_y, t_n)}]}{\sqrt{\sum_{i=1}^{m}[f(h_x, t_n, s_i) - \overline{F(h_x, t_n)}]^2}\sqrt{\sum_{i=1}^{m}[f(h_y, t_n, s_i) - \overline{F(h_y, t_n)}]^2}} \tag{3.1}$$

The equation calculates the Pearson correlation coefficient between the traffic pattern of hashtag h_x and that of hashtag h_y. Algorithm 3 guarantees that the input keyword set for the next time frame t_{n+1} is a list of hashtags where $h_k \in H(t_n)$ with traffic pattern that highly correlated to that of initial keywords. For example, *#100aday* is a trending hashtag during the 2012 London Olympic Games, but irrelevant to the event. It is detected as a keyword by TF-KwAA, but successfully excluded in TP-KwAA because of its low correlation to the initial seeds.

3.3.3 Content Similarity based Approach (CS-KwAA)

As illustrated in section 3.6.1, applying the adaptive crawler with TP-KwAA achieves a better result than using TF-KwAA since the probability of a noisy hashtag becoming keyword is reduced. However, with the increasing number of tweets and events crawled, TP-KwAA also shows its limitation. Specifically, it is not stable enough to identify event-related keywords in a consistent way under all kind of events. Sometime, the dataset it crawled contains a large amount of irrelevant tweets, especially when something trending happens. This is due to the misleading of Twitter Streaming API on estimating the frequency of top hashtags. When the event is discussed extensively on Twitter, the volume of tweets about the event is more likely to exceed the 1% limit. With non-uniform sampling used by Twitter Streaming API, the correlation between frequency counts is less accurate [80]. Although it can recover without human intervention once the trending over, this behaviour is not desirable and leads to keyword set contain high degree of noise.

As a result, this research proposes a third version of adaptation algorithm, CS-KwAA, which relies on determining the tweets similarity between hashtags. The assumption of this approach is that the text content of a collection of tweets, which represents of the

recent activity of any potential new event related hashtag, should be textually similar to that of a collection of tweets containing the initial keywords. In order to build this representative tweet collection or hashtag profile, this KwAA collects previous tweets posted that contain the hashtag. One hashtag thus can be represented by a Hashtag-based TF-IDF vector that is constructed using all the tweets in it profile. The similarity between a new hashtag and initial keyword can be measured by computing the similarity between their Hashtag-based TF-IDF vectors. Therefore, hashtags with high similarity to the initial keywords are considered as new search terms.

3.3.3.1 Hashtag-based TF-IDF vector

In order to identify as many event-related documents as possible, a measurement to evaluate their relevance to the event is necessary. The majority of existing research uses the bag of word model on tweets with TF-IDF vector. [114, 115, 116].

Term Frequency - Inverse Document Frequency (TF-IDF), is a document vectorizor that statistically measures how important a word is to a document in a collection or corpus. When building the TF-IDF vector for tweets, the conventional approach regards each tweet post as a single document [20, 32]. As a result, the term frequency (tf) value of a term t in a tweet is calculated by equation 3.2, where N_t is the number of times term t appear in tweet d. The inverse document frequency (idf), which measures whether a term frequently appears across the whole corpus D, is calculated based on the total number of tweets in the corpus (N_D) and the number of tweets contains term t ($N_{t \in D}$). However, when the term t is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator by plus one, as show in equation 3.3.

$$tf(t, d) = N_t \tag{3.2}$$

$$idf(t, D) = \log \frac{N_D}{N_{t \in D} + 1} \tag{3.3}$$

The product of $tf(t, d)$ and $idf(t, D)$ above is the TF-IDF value of term t. Considering that a tweet update is made up of multiple terms $(t_1, t_2, ..., t_n)$, the TF-IDF vector of

tweet d, i.e. $tfidf_d$, consists of all the TF-IDF value of all the appearing terms, as show in equation 3.4.

$$tfidf_d = \{tfidf_1, tfidf_2, ..., tfidf_d\},\ where\ tfidf_d = tf(t,d) \times idf(t,D), (d = 1, 2, ..., n) \tag{3.4}$$

Although the conventional usage of TF-IDF is proved to perform well in structured, long paragraph article, its accuracy in short and noisy sentences, such as tweets, is still not good [117]. Tweets are always informal, irregular and with many spelling errors and abbreviated words. Moreover, the narrative expression format and mixed languages also bring difficulties on the application of TF-IDF to short length content like tweets. In order to judge the content similarity of one hashtag to another hashtag, rather than comparing the basic similarity of the TF-IDF vector of individual tweets that contain the hashtags, this research proposes building longer profile description D_h that better describes the characteristics of the hashtag, i.e. hashtag-based TF-IDF vector. The procedure for constructing the hashtag-based TF-IDF is shown in Figure 3.6.

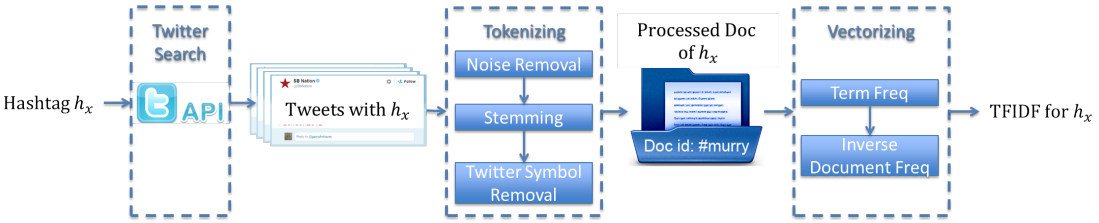


FIGURE 3.6: Construction Procedures of Hashtag-based TF-IDF vector

When a candidate hashtag h_x is identified, a document consisting of a collection of the last 100 historical tweets containing that hashtag is built. All the tweets for building the document are retrieved from the Twitter Search API[3].

Tokenizing the document is the second step in this TF-IDF vector construction task. Each hashtag-based document is analysed by an original Twitter Analyser based on Lucene[4]. This research follows some common pre-processing approaches as listed below:

[3]Documentation for Twitter Search API: `https://dev.twitter.com/rest/public/search`
[4]Apache Lucene: `https://lucene.apache.org/`

- **Noise Removal**: A raw Twitter post always contains information that doesn't provide content for the event detection, such as punctuation, emoticon and stop-words. In order to extract bags of meaningful terms from the original tweets, all datasets are processed by the same text analyser to remove the punctuations and stop-words. In addition, the redundant repeat characters in the original post are processed to convert to the most similar word in the corpus. For example, words like yeeeeeeeaaaaaaaah will be converted into yeah.

- **Stemming**: Word stemming, which converts the derived words into their root, is a common step in text analysis. In the extremely noisy Twittersphere, this action reduces the dimension of feature space by mapping the various inflected words to the same stem. In this research, the Mahout[5] implementation of Porter stemming algorithm is employed for word stemming.

- **Twitter Symbol Removal**: Apart from the noisy characters removal and word stemming, all the raw tweets in the datasets are processed with a Twitter specific analyser. This analyser is designed to remove inherent Twitter symbols (in section 2.3.1.2), such as user mention, retweet symbol and URL link. Though these symbols are useful for social relationship analysis [76], they made no contribution or even bring negative impact in the Twitter event detection. However, the # hashtag symbol was kept since it is not only the indication of the topic but also useful for the adaptive keyword identification.

The remaining words are tokenized into uni-grams[6] and used as input for the construction of hashtag-based TF-IDF vectors. Unlike the conventional tweet level TF-IDF vector which severely suffers from the sparsity issue, the TF-IDF vector built with a collection of tweets becomes more dense and meaningful for comparison. The aggregated document for each hashtag is much more descriptive and informative when compared with the short single post. This is because this kind of hashtags pooling strategy better describe the topic they related to and provide additional dimension to be calculated [118].

[5] Apache Mahout: `https://mahout.apache.org/`

[6] an n-gram is a contiguous sequence of n terms from a given sequence of text, where uni-gram indicating that n=1

Same as the first step in TP-KwAA, Algorithm 2 is executed at the beginning to generate an array of the potential hashtags, represented as $H(t_n)$. Then, by following the vector construction method described in section 3.3.3.1, this algorithm builds the hashtag based TF-IDF vector for each $h_x \in H(t_n)$. The similarity between hashtag h_x and an existing keywords $h_y \in H_{seed}$ are quantified by the cosine distance between their TF-IDF vectors. Hashtags $\forall h_x \in H(t_n)$ that is distant from h_y below the pre-defined threshold $Thres$ are considered to be related to the event. Thus are selected as new keywords for querying Twitter API during next time frame. The pseudocode for Algorithm 4 gives a more detailed explanation:

Algorithm 4 Content Similarity based Keyword Adaptation Algorithm (CS-KwAA)

Require: $H_{seed} = H(t_n) \cup H_{fin}(t_{n-1}), H_{fin}(t_n) = H_{BL}$
1: **Execute Algorithm 2**
2: **for** $\forall h_x \in H(t_n)$ **do**
3: **for** $\forall h_y \in \{H_{seed} \cup H_{fin}(t_n)\}$ **do**
4: **if** $sim(F(h_x, t_n), F(h_y, t_n)) > Thres$ **then**
5: **if** $h_x \notin \{H_{seed} \cup H_{fin}(t_n)\}$ **then**
6: add h_x to $H_{fin}(t_n)$
7: **end if**
8: **end if**
9: **end for**
10: **end for**

In this algorithm, cosine similarity is employed to quantify the differences between each pair of candidate hashtags. This distance measurement is widely used in text mining since it is proved to be an efficient and effective text similarity measurement [119]. Consequently, the content similarity between two hashtags h_x and h_y, i.e. sim in the above algorithm 4, can be measured by the cosine distance between their TF-IDF vectors $tfidf_{d1}$ and $tfidf_{d2}$, which is defined in equation 3.5.

$$sim(h_x, h_y) = \cos(tfidf_{h_x}, tfidf_{h_y}) = \frac{tfidf_{h_x} \cdot tfidf_{h_y}}{||tfidf_{h_x}|| \ ||tfidf_{h_y}||} \tag{3.5}$$

By employing the above comparison, the algorithm captures additional event-related trending hashtags and introduces additional event contents. In fact, the recent 100 historical tweets also introduce the temporal feature latently as their timestamps are closest to the time of calculation. This research heuristically determine $Thres = 0.5$

since the similarity of tweets shows extreme value: either close to one or close to zero [52].

3.4 Evaluation Approaches

When determining the relevance of the retrieval result, manually assignment of event labels is the common practice [84, 109, 120]. However, this is not realistic in practice for modern information retrieval, as the number of relevant instance can be extremely huge. Although it is possible to label all the identified keywords, the volume of retrieved tweets is too huge to access their relevance. Some researcher explored the low-cost evaluation techniques in IR to only access the precision and recall with top ranked items [121]. Retrieval methods evaluated with these approached aim at providing limit amount of highly relevant event tweets, which is different from the interests of this research: to expand the coverage of event content by retrieving comprehensive set of event tweets. Consequently, a semi-automatic evaluation approach is employed. This approach assesses the relevance of retrieved tweets by the event related keyword is explored. Though the evaluation process requires the manually relevance assessment, this is to examine the reliability of the proposed adaptive crawling and KwAAs, thus doesn't prevent the proposed algorithms and model running in fully automatic way. The following subsection first gives the rationale of the evaluation approaches (in section 3.4.1), then details the rules used to quantify the event relevance of a hashtag (in section 3.4.2) and the automated way to determine tweet's event relevance (in section 3.4.3).

3.4.1 Preliminary

The fundamental assumption made by the majority of the Twitter research work is based on a common hypothesis: a single tweet only talks about one topic [32, 70]. Since hashtags can be considered as the topic indicator of a tweet [113], the hypothesis aforementioned is updated for evaluating the three proposed KwAAs:

Assumption 3: a tweet only talks about one topic, and can be described by the hashtags it contains.

According to Assumption 3, the relevance between a tweet and an event can be determined by the relevance between its hashtags and the event of interests. As a result, rather than assessing the relevance between the complete content of a tweet and the event, the evaluation can be heuristically completed by examining the relationship between tweet hashtags and the event.

3.4.2 Hashtags Labelling

In order to determine the event-relevance of a tweet by its hashtags, the hashtags relevance to the event needs to be assessed first. To distinguish whether a hashtag is related or non-related to the event, this research labels the hashtags using human efforts. Three independent participants are involved in this labelling process, and all provides with a strategies labelling table (as shown in Table 3.1). All hashtags that appear in the evaluation period of an event are manually classified by the participants into corresponding categories. The final result is based on the majority choice of the three independent participants. If all three participants don't reach agreement on a hashtag, it will be labelled as "Not Known" C3.

TABLE 3.1: Hashtag Categorization and Grading Strategy

Abbr	**Hashtag Category**	**Specification**	**Score**
C1	Related	hashtags that contain the keywords about the event of interests, or name entity that are involved in the event	+2
C2	Possibly-related	hashtags that are more general to the event of interests but still related to the event of interests	+1
C3	Not known	hashtags that are ambiguous or hard to assign to a category	0
C4	Non-keyword	Hashtags that have not been selected as keywords	-1
C5	Non-related	hashtags showing no particular relationship with the event	-2

Hashtags in different time periods are labelled according to how closely they are related

to the event. Take the 2012 London Olympic Games [7] as an example: "*#2012olympic*" is definitely related to the event, while hashtag "*#harrypotter*" is difficult to classify. It could be related since the characters in Harry Potter was used in the performance of the opening ceremony. However, it is likely to introduce information irrelevant to the Olympic Games. According to the proposed grading strategy, this kind of hashtags are classified as possibly-related.

3.4.3 Automatic Tweets Classification by Hashtags Categories

In this step, based on the grading strategy in Table 3.1 tweets are classified into either event relevant or event irrelevant based on the hashtags it contains. Each hashtag is assigned with a score and the final grade of a tweet is calculated by summing the scores of its hashtags.

By using this strategy, tweets with a grade higher than 0 are classified as event relevant content, and those less than or equal to 0, as event irrelevant content. However, for tweets which only contain initial keyword(s) but no hashtag, they are classified as event irrelevant content even though it contains the initial keyword. As a result, a special grading rule is applied to tweets which contains the initial plain text keyword(s): These tweets are scored as +2. The grading strategies listed in Table 3.1 help to identify event irrelevant tweets even if it carries event-related hashtags. For example, "*#TVHighlights: July 2012 #olympics #glee revenge #onceuponatime #newgirl #idol #antm #xfactorph #asap2012*" is obviously a non-related tweet though it carries #olympic. The total grade is -6 because the positive score introduced by #olympic is cancelled out by other hashtags. On the other hand, this strategy also identifies related tweets when it has non-related hashtags. "*Phelps came from behind to help USA win the gold in the 400 medley relay. #justwow #London2012 #GetGlueHD #Olympics*" is a related tweet which contains hashtags in C1 (#London2012 and #Olympics), C3 (#GetGlueHD) and C4 hashtags (#justwow). It is classified as relevant with a grade +1.

Consequently, the event datasets are divided into two different parts: one that contains

[7] 2012 London Olympic Games: `https://en.wikipedia.org/wiki/2012_Summer_Olympics`

the event relevant content while the other one consists of only the event irrelevant content. Finally, by comparing the proportion of event relevant and event irrelevant tweets across those two parts, it is possible to quantify both the amount of event relevant information and the level of noise introduced by the proposed adaptive crawlers.

3.5 Parameter Tuning

The thresholds in the TP-KwAAs need to be determined before running the crawlers. This section discusses the configuration of the parameters used in the proposed algorithm. In this section, the correlation measurement cor, the thresholds for determine whether the traffic pattern of two hashtags are similar $Thres_1$ and $Thres_2$, as well as the length of time frame t_n are tuned by datasets retrieved during 2012 London Olympic Games. The experiment is based on the tweets from the 2012 London Olympic Games baseline dataset. Three time periods covering the opening ceremony, the women's badminton final and the closing ceremony were chosen for this investigation. Although the two parameters are trained on events relating to London Olympic Games, the traffic pattern and content characteristics of selected periods are different. Therefore, the parameters tuned with this experiment can be generalised to other event. Table 3.2 presents all the testing parameter and their values for this experiment.

TABLE 3.2: Parameters List and Testing Values for TP-KwAA

Parameters	**Value**		
Correlation Coefficient	Pearson's r	Kendall's τ [122]	Spearman's ρ [123]
$Thres_1$	$(0, 1)$ *by* 0.1		
$Thres_2$	$[0, 1)$ *by* 0.05		
Time Frame Interval	5 mins	10 mins	20 mins

In order to retain the hashtags that are relevant to the Olympics while minimising the irrelevant ones, this research explores different combinations of the above parameters in reference to the proportion of each type of hashtag that are kept after filtering. In detail, the relevance of $h_k \in H(t_n)$ to the Olympic event is labelled according to the strategy in 3.4.2 and the retained ratio of all four categories (defined by equation 3.6)

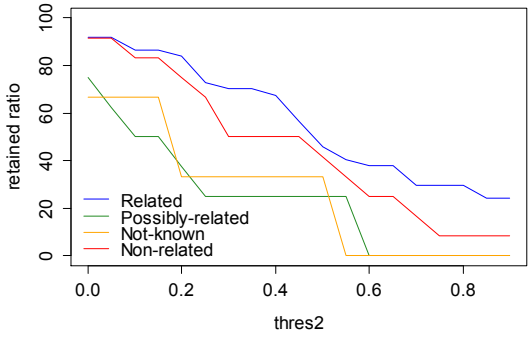
is calculated.

$$\text{retained ratio} = \frac{\text{number of hashtags after filter}}{\text{number of hashtags in total}} \tag{3.6}$$

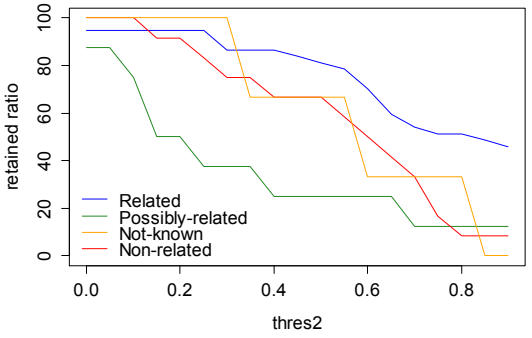
By applying correlation measurement to their frequency count $F(h_k, t_n)$, hashtags with low correlation to the identified keywords are filtered out while others are kept. This research then examines the proportion of hashtags that are retained (i.e. retained ratio) in each category. The ultimate goal of this experiment is to find the threshold values which guarantee a relative high retained ratio for the event-related hashtags and a low value for non-related ones.

Three commonly-used correlation coefficients, including Pearson, Kendall and Spearman, are employed in this test. To determining the threshold $Thres_1$ and $Thres_2$, this research adopts a single variable approach [124]. In this case, assuming that threshold $Thres_1$ is fixed, threshold $Thres_2$ is changed gradually in each single test. Thus, the value of $Thres_2$ can be determined based on the results of the group of tests with same $Thres_1$ value. Based on the Assumption 2.1, the range $[0, 1)$ for positive correlation is chosen. In this experiment, the value 1 is excluded due to the reason that the total positive correlation is rare in the given scenario. As listed in Table 3.2, this research also explores the retained ratio for three different time intervals. Since the time interval needs to be further divided to generate hashtag count sequence for calculating correlation (in equation 3.1), the shortest time interval need to provides enough traffic characteristics for comparison. A longer time interval gives better traffic characteristics but extends the waiting period for new keywords identification. As a result, 5 minute interval is used as the shortest time frame, and a 20 minutes interval is used as the max time frame in this experiment. Figure 3.7 illustrates the proportion of hashtags, with different relevance to the event that is retained given the different threshold values. Note that while more than two hundreds of figures are generated for all the evaluation periods, this thesis presents figures that justify the selection of the parameter values. These 5 figures are produced with the data from the same time period. They illustrates the best performance can be achieved with all the possible combination of the parameter values.

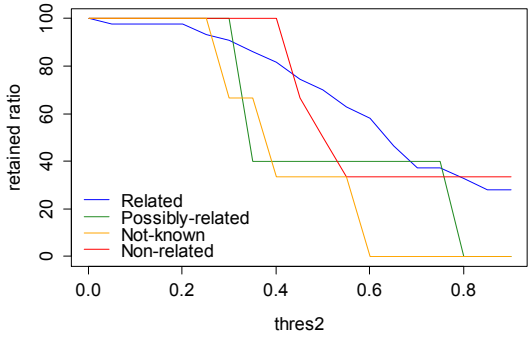
As can be observed from Figure 3.7 (a) - (b) - (d) , Pearson's correlation coefficient



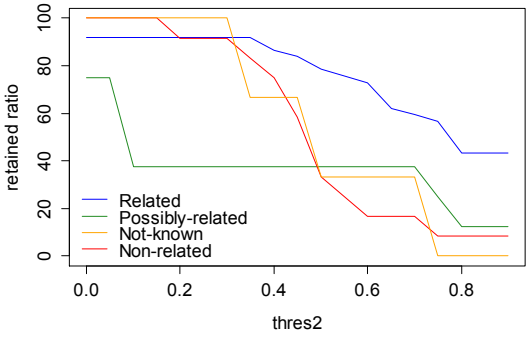
(a) Kendall for 10 Minutes Interval



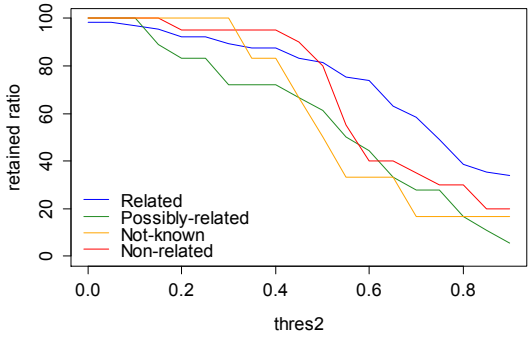
(b) Spearman for 10 Minutes Interval



(c) Pearson for 5 Minutes Interval



(d) Pearson for 10 Minutes Interval



(e) Pearson for 20 Minutes Interval

FIGURE 3.7: Parameter Tuning for TP-KwAA

(a) - (b) - (d) comparison between correlation measurements (Kendall's τ - Spearman's ρ - Pearson's r) with 10mins interval.

(c) - (d) - (e) comparison between different time interval (5mins - 10mins - 20mins) with Pearson's r;

gives the best filtering results among the three correlation measurements. This can be observed from the apparent gap of the retained ratio between related keywords and other keywords, especially the non-related keyword. The Spearman's rank correlation coefficient also help to filtered more non-related hashtags. However, the gap between retain ratio of related hashtags and non-related hashtags is less obvious (according to Figure 3.7 (b)). The Kendall's rank correlation gives the worst performance. As shown in Figure 3.7 (b), it almost filters the same amount of event related and non-related hashtags.

The heuristic exploration on $Thres_1$ shows a common pattern for all the correlation coefficients. A low $Thres_1$ (less than 0.4) is inadequate to filtered out non-relevant hashtags. While a value higher than 0.7 always results in a quick drop of retain ratio on all types of hashtags, sometimes with no hashtags left for the hashtags recapturing with $Thres_2$. Based on the observation of multiple experiments over different time periods, all the correlation measurements achieve the best performance around $Thres_1 = 0.5$. The changing of $Thres_2$ also impact the retain ratio in the same way as the $Thres_1$ does. In order to select the value for $Thres_2$, the aim is to find a value that maximise the gap between the red and the blue line in Figure 3.7. The maximum distance is achieved when $0.6 \leq Thres_2 \leq 0.8$. Considering that the other aim is to maintain a high retained ratio for related hashtags, and the performance over other time periods, $Thres_2$ is set to be 0.7.

The effect of changing time interval to the filtering results can be observed from Figure 3.7 (c) - (d) - (e) . When the time interval is 5-minutes, Pearson coefficient filters more non-related hashtags than related hashtags for only a narrow range of $Thres_2$. The difference is about 20% maximal when $Thres2 = 0.65$ (in Figure 3.7 (c)), which is less than that (about 60% when $Thres2 = 0.6$) for 10-minutes time interval (in Figure 3.7 (d)). The retained ratio doesn't increase when the time interval grows to 20 minutes, the maximal difference is about 40% when $Thres2 = 0.6$ (Figure 3.7 (e)). Although the plots of Kendall and Spearman for their 5-minutes and 20-minutes time interval are not presented, the pattern is the same as that shown in Pearson plot. As a result, a 10-minutes time interval is selected.

3.6 Performance Evaluation Results

The aim of this evaluation process is to verify that the adaptive crawling model performance well in retrieving extra amount of event-related information across different types of events. This section presents the evaluation results with two different experiments. In details, with the aim to get a comprehensive understanding of the proposed adaptive crawling model and the three KwAAs, the evaluation of the proposed KwAAs is done from two aspects: the comparison between different KwAAs (in section 3.6.1), and their performance over different type of events (in section 3.6.2). Finally, the performance of proposed adaptive crawling and the characteristics of the KwAAs is discussed and summarised (in section 3.6.1.4).

3.6.1 Comparison across KwAAs

The main different of the adaptive crawler and the baseline crawler is the Keyword Adaptation Component (in Figure 3.4). This sub-section compares between the baseline crawler (the non-adaptive crawling model in section 3.2.1) and the adaptive crawlers with different KwAAs (in section 3.3) by using a real-world event as the evaluation dataset. The following sub-sections first present the dataset characteristics of 2013 Glastonbury Music Festival[8] used for this analysis (in section 3.6.1.1). Then, the results on the identified keywords (in section 3.6.1.2) and retrieved tweets (in section 3.6.1.3) are given. Finally, the performance of the KwAAs is discussed (in section 3.6.1.4). While an initial evaluation of the Olympic Datasets was presented in the published papers [125], these new datasets give more insights about the proposed KwAAs.

3.6.1.1 Event Dataset

Datasets for this evaluation are retrieved during the 2013 Glastonbury Music festival period. Four crawlers are deployed during the collection: the baseline crawler, and

[8]Glastonbury Festival is the UK's largest music festival that hosts contemporary performing arts, including but certainly not limit to music, dance, comedy and etc. `https://en.wikipedia.org/wiki/Glastonbury_Festival_2013`

three adaptive crawlers equipped with the different KwAAs (TF-KwAA, TP-KwAA or CS-KwAA). Only"**Glastonbury**" is used as initial keyword for all four crawlers. As a result, four separate datasets are collected by fetching tweets from the real-time Twitter Streaming API .

Table 3.3 describes the tweet volumes harvested during the collection period *2013-06-28, 19:00* to *2013-07-01, 07:00*. In this experiment, the parameters in each algorithm are

TABLE 3.3: Tweet Volume Generated by Different Crawling Approaches (Glastonbury)

	Baseline	TF-KwAA	TP-KwAA	CS-KwAA
Tweet Count	550,417	10,433,355	2,472,953	753,027
Unique Tweet	6 (0.00%)	9505198 (91.10%)	1206464 (48.78%)	115091 (15.28%)

configured based on the tuning results (described in section 3.5). In order to compare all the KwAAs fairly, the time frame is set to be 10 minutes for all the KwAAs. The entire collection period lasts for 60 hours and more than half million tweets are collected from the baseline crawler alone. The tweets datasets retrieved through the adaptive crawling approach with different KwAAs are even larger: the TF-KwAA dataset contains the largest amount of tweets, 20 times that of the baseline dataset'.

Table 3.3 also details the unique tweets (both event relevant and event irrelevant) that don't show up in any of the other three datasets. There are 6 unique tweets in baseline dataset. Since all of the crawlers should be able to collect tweets contain Glastonbury, this result is due to the other crawlers hit the rate limits. The TF-KwAA dataset is not only the largest one but also the most unique one: more than 90% of tweets from it are missed by other crawling approach. However, by doing a simple calculation, it is shown that TF-KwAA accumulates 2900 tweets in every second, almost the rate limit of Twitter streaming API. This statistic means that much of the baseline traffic could be missed in TF-KwAA dataset as space is occupied by other keywords. It is observed that both the TP-KwAA dataset and CS-KwAA dataset also contain lots of unique traffic. This is likely an indication of the differences introduced by collecting of new keywords found by KwAAs. In order to investigate this hypothesis, as well as the composition of the additional keywords, Table 3.4 is produced to give an overview about the retrieval

keywords. In this table and the rest of this thesis, the keywords are counted distinctly.

TABLE 3.4: Number of Keywords Identified by Different Crawling Approaches (Glastonbury)

	Baseline	TF-KwAA	TP-KwAA	CS-KwAA
Distinct Keyword Count	1	2908	2876	407
Unique Keyword	0 (0.00%)	1919 (65.99%)	1751 (60.88%)	199 (48.89%)

In other words, keywords occurring multiple times during the evaluation period are regarded as the same.

The baseline dataset is retrieved by using a single keyword "Glastonbury", so there is only one keyword for the baseline dataset. Other datasets are generated by a set of keywords which are identified by different KwAAs. According to the statistics in Table 3.4, the number of distinct keywords identified varies between the three KwAAs. It is clear that the TF-KwAA and TP-KwAA keyword sets are filled with a lot of new distinct hashtags, while the number of distinct keywords identified by CS-KwAA is much less (less than seventh of the other two). The result shows that the large proportion of unique tweets, shown in Table 3.3 are the results of these different keywords. The keywords identified by TF-KwAA is most different to the other ones': more than 65% of keywords don't show up in the keyword lists of the other KwAAs. On average, 50% of the keywords are distinct when using different KwAAs. Even for the least unique keyword set produced by CS-KwAA, almost 200 unseen keywords are shown in the list, i.e. 2 distinct keywords in each time frame. A further investigation about the composition of all the keywords by different algorithms is presented in the Table 3.6.

Figure 3.8 plots the total traffic volume over the collection period, where the count is sampled every 5 minutes. As shown in the figure, the number of tweets from TF-KwAA dataset is always below 15000, i.e. 3000 tweets/min no matter what keywords the crawler runs with. A test over Twitter streaming API in the same period indicates that the upper limit is about 3000 tweets/min. As shown, the traffic volume of TP-KwAA dataset sometimes also reaches the rate limits. Neither the Baseline nor CS-KwAA reach the rate limit. Furthermore, both show similar trends, but differ in volume: CS-KwAA collecting larger amount of tweets during some periods.

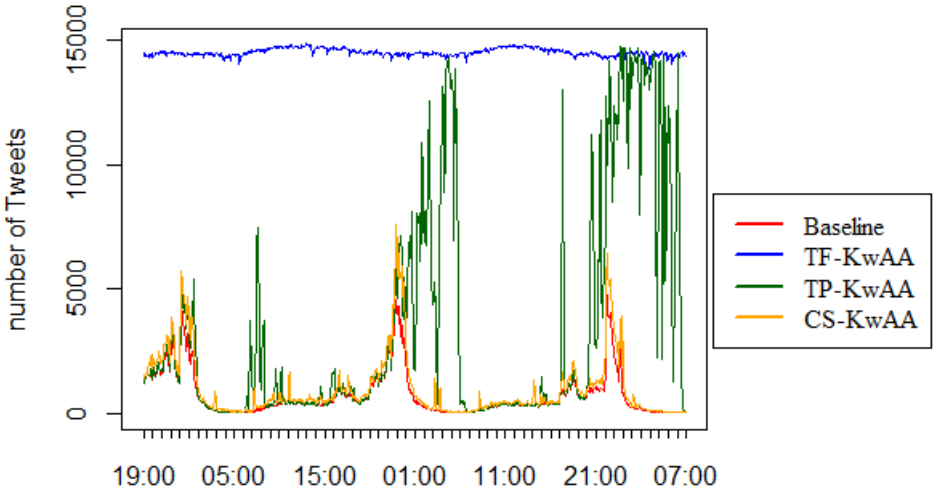


FIGURE 3.8: Tweet Volume for 2013 Glastonbury Music Festival

A subset of the Glastonbury data is selected for the evaluation, i.e. timestamp within the period of *2013-06-29, 08:00* to *2013-06-30, 04:00*. This period covers the typical stages of a single day festival: from the tranquil morning, the exciting evening performance and the end of people's exiting. This period is a good evaluation for the crawlers since it covers both high and low activity times in the Twitter stream related to the event. The traffic pattern of all the crawlers during the selected period is shown in Figure 3.9.

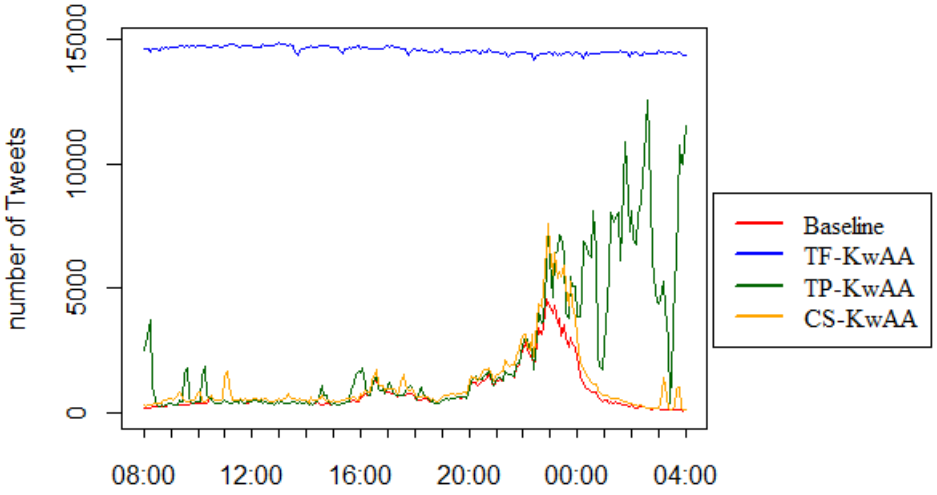


FIGURE 3.9: Tweet Volume for 2013 Glastonbury Music Festival (Evaluation Period)

According to the tweet volume of Baseline dataset in Figure 3.9, the first apparent increasing happens at 16:00 on the evaluation day, while the highest traffic period starts at night from about 20:00, and reaches the peak at about 23:00, then quickly drops

at the midnight. This is because most of the Glastonbury music performances start at the afternoon and end before midnight. When the show finishes, people still post their comments and opinion about the past performance, which leads to a short tail, their comments on the performance can sometimes take an hour to fade out.

3.6.1.2 Event-relevance of Identified Keywords

To evaluate the performance of different KwAAs, this research first examines the event relatedness of all keywords identified by the KwAAs, It then looks at the common evaluation metric, precision, recall and F1 score (in section 2.2.3). Follows the labelling steps in section 3.4.2, the retrieval keywords are categorised according to the criteria in Table 3.1. Table 3.5 gives some examples for the labelling process.

TABLE 3.5: Hashtag Categorization and Grading Strategy

Abbr	**Hashtag Category**	**Example**	**Score**
C1	Related	hashtags contain "glastonbury", band names or song names that appear during the festival (#glaston2013)	+2
C2	Possibly-related	media channel that in charge of the festival broadcasting, emotional hashtags about the ongoing affairs (#nextyear)	+1
C3	Not known	non-English hashtags, or hashtags that can bring both relevant and irrelevant tweets (#music)	0
C4	Non-keyword	hashtags that have not been selected as keywords (#100aday)	-1
C5	Non-related	hashtags that refers to other events or typical Twitter topic (#teamfollowback)	-2

As shown in Table 3.6, the retrieval keyword set from TF-KwAA is the noisiest one as it's nearly full of C4 keywords. Although there are still some C1 and C2 keywords for the TF-KwAA keywords list, the huge volume of C3 and C4 keywords can easily pollute its dataset. The keywords list generated by TP-KwAA contains more C1 and C2 keywords which help to introduce more event relevant tweets. However, though the TP-KwAA reduces the noisy keywords (C3 and C4) to a lower level than that of TF-KwAA, it still introduces far too many irrelevant keywords (C3) when considering the adopted ratio to

TABLE 3.6: Keyword Categories by Hashtags Labelling for 2013 Glastonbury Festival (Evaluation Period)

	Baseline	TF-KwAA	TP-KwAA	CS-KwAA
Related (C1)	1	15	74	121
Possibly-related (C2)	0	16	40	32
Not known (C3)	0	500	167	0
Non-related (C4)	0	1360	636	41
Total	1	1891	917	194

the potentially useful keywords (C1 and C2): more than 70% of keywords are irrelevant to the event. Among the three versions of KwAAs, CS-KwAA performs best for the 2013 Glastonbury Music Festival. There are 121 C1 keywords and 32 C2 keywords, i.e. more than 78% of all keywords, are highly related to the event. The rest of others are C4 keywords, which occupy less than 22% of the keywords list. The number of total keywords in this table is less than that in Table 3.4 because only the keywords identified in evaluation period (2013-06-29, 08:00 to 2013-06-30, 04:00), rather than the whole collection period (2013-06-28, 19:00 to 2013-07-01, 07:00), are considered here.

With the general information for the whole evaluation period, a further exploration on the composition of identified keywords is done by analysing the keyword categories of some sample periods. For an hour period, one time interval (out of the total six) is randomly selected for hashtag labelling. The commonly used evaluation metrics - precision (P) , recall (R) and F_1 score (F_1) are adopted to assess the keyword identification. These are widely used in the existing Tweets retrieval research [34, 48, 50]. In this keyword relevance evaluation process, precision refers to the proportion of identified keyword that are relevant to the event, while recall is measured by the proportion of event relevant hashtags that are existed in the last time frame. Namely, only the hashtags in the last t_{n-1} time frame is used to calculate recall. This is because the identification of all the event-related keywords for a time frame t_n in the Twitterverse can be unrealistic. While the precision and recall is measured by equation 2.1 and 2.2 respectively, the F1-score, as a special case of F-measure, equally weights the precision and recall. By making $\alpha = 1/2$ or $\beta = 1$ in equation 2.3, the formula for F1-score can be simplified to equation 3.7.

$$F_1 = 2 \cdot \frac{P \times R}{P + R} \tag{3.7}$$

Table 3.7 lists the number of relevant keywords (either C1 or C2), precision, recall and the F1 score for keyword identification during some of the sample periods. The precision,

TABLE 3.7: Precision and Recall of Keyword Identification for 2013 Glastonbury Festival (Evaluation Period)

	TP-KwAA				CS-KwAA			
	No. Key[†]	P (%)	R (%)	F1 (%)	No. Key	P (%)	R (%)	F1 (%)
06-29 08:00	1	4.8	25	8.0	6	67	86	75
06-29 11:51	2	67	25	36	10	83	83	83
06-29 13:11	4	67	22	33	12	100	92	96
06-29 14:11	1	33	14	20	9	75	82	78
...				... ...				
06-29 20:12	4	100	15	40	27	90	73	81
06-29 23:12	25	57	54	56	33	81	65	72
06-30 02:33	2	1.5	40	2.9	12	80	80	80
06-30 03:53	2	5	67	9.3	10	71	83	77
Average	**4.6**	**53**	**31**	**29**	**16**	**82**	**83**	**81**

[†]sum number of keywords in event related or possibly-related category

recall and F1 score are calculated based on the keyword number listed in the table. In this part, only the results for TP-KwAA and CS-KwAA are presented since the keyword set for TF-KwAA is too noisy to be manually examined.

Based on the data from Table 3.7, it can be observed that the precision of TP-KwAA can be very low (only 1.5%), while sometime achieve the 100% accuracy. The lowest precision for TP-KwAA is observed during the period without music performance, where most of the identified keywords are not relevant to the event. The high precision of TP-KwAA sometimes follows with a low recall, as that shown in the result of "06-29 20:12". This means that the algorithm only detects a few event keywords (normally under 5) while many of event relevant hashtags are missed. While only 4 event-related keywords are identified by TP-KwAA at 20:12, 27 keywords are identified by CS-KwAA. This difference indicates that a large amount of information captured by CS-KwAA is lost by TP-KwAA. Another important issue is that the sudden outbreak of keywords number makes this algorithm more vulnerable once the rate limits are reached.

The result of CS-KwAA is more optimistic. The performance of this algorithm is very stable since all three metrics stay around at 80%. The lowest value of each metric is 67%, 65% and 72% respectively. In addition, the examination on keyword set shows

that number of useful keyword in each time frame for TP-KwAA varies a lot (from 1 to 25), while the value for CS-KwAA doesn't fluctuate much. In other words, CS-KwAA can constantly pick useful event keywords no matter during performance period or not.

3.6.1.3 Event-relevance of Retrieved Tweets

After the keyword labelling, tweets in all the datasets are classified based on the total score of their hashtags'. As described in section 3.6.1.2, hashtags in the four datasets are combined and labelled. As a result, identical tweets in the baseline dataset and the other three adaptive datasets receive the same classification and thus get the same score. Table 3.8 demonstrates the results of tweet classification.

TABLE 3.8: Event Relevance of Retrieved Tweets for 2013 Glastonbury Festival (Evaluation Period)

	Baseline	TF-KwAA	TP-KwAA	CS-KwAA
Event Relevant Tweets	201,683 (97.64%)	191,096 (5.47%)	232,797 (42.60%)	260,897 (90.82%)
Event Irrelevant Tweets	4,875 (2.36%)	3,301,592 (94.53%)	326,814 (58.40%)	26,356 (9.18%)

The tweets classification result of TF-KwAA dataset further illustrates the observation in previous section that TF-KwAA dataset is the noisiest. A great proportion of non-related keywords (more than 70%) results in only 5.47% event relevant tweets. Due to the Twitter rate limit, the greater the amount of retrieval keywords, the less traffic can be fetched by every keyword. As the event irrelevant keywords occupy a large amount of space, the proportion of event relevant tweets is reduced to even fewer amounts. This issue is somewhat alleviated in the TP-KwAA dataset, more than 42% of tweets are related to the event. However, this is still far from the baseline standard. As can be seen from Figure 3.8, TP-KwAA shows similar behaviour as the TF-KwAA in the later stages, after 00:00. As shown in the event relevant tweets versus event irrelevant tweets plot in Figure 3.10, although the TP-KwAA dataset maintains the proportion of noise at a low level until 2013-6-30, 00:00, it is then quickly polluted with event irrelevant tweets due to the large amount of C4 keywords.

One phenomenon shown by Figure 3.10 is that there are event irrelevant tweets in

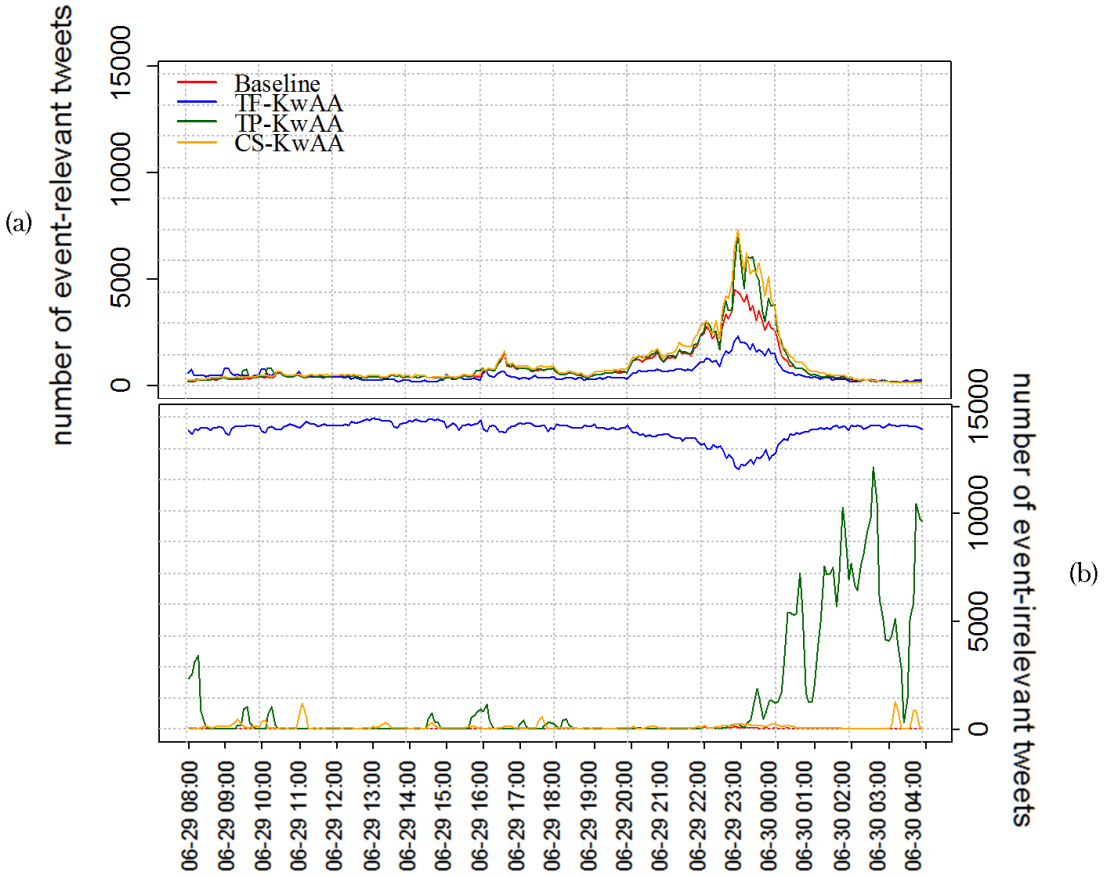


FIGURE 3.10: Event-relevant tweets (a) versus Event-irrelevant Tweets (b) for 2013 Glastonbury Festival (Evaluation Period)

the baseline dataset. Even though the keyword "Glastonbury" is highly specific, the total score can be negative due to the impact of hashtags which are labelled as C4 and C5.In order to spread trending topics and hashtags, the spam tweets always carry many independent hashtags, such as *Some great T.V. On this this weekend! #wimbledon #britishirishlions #tourdefrance #glastonbury #grandprix #europeantourgolf.* These tweets become one of the major sources of the irrelevant tweets in baseline dataset, but can be identified and filtered out by the grading strategy in Table 3.6. There are also other types of event irrelevant tweets that might be wrongly classified by the grading strategy. For example, tweet "*Who needs #glasto when you've got this 3 man band and a field of Whittlebury campers?*" will be classified relevant since it contains event-related keyword "#glasto". However, the tweet is not that relevant to the Glastonbury Music Festival as it doesn't reveal any underlying stories of the festival. As only a small proportion of this type of noise appears in the datasets, filtering them has little impact

on the overall classification results.

The TF-KwAA dataset contains large amount of noises (i.e. event irrelevant tweet) according to the tweets classification result in Table 3.8, thus results in a fairly flat line with a very high value for the event irrelevant tweets. Also, since the space is occupied by a lot of noisy tweets, the amount of event relevant tweets is much less than the other datasets, even the baseline. The CS-KwwA dataset not only collects additional event relevant tweets but also maintains a low ratio for the event irrelevant tweets. The composition of this CS-KwAA dataset is similar to the baseline one. This figure demonstrates that both TP-KwAA and CS-KwAA collect a significant amount of event relevant tweets. However, unlike TP-KwAA, CS-KwAA keeps itself from being polluted by noise. As shown in (b) Figure 3.10, the noise level is much lower than the TP-KwAA.

For a more intuitive view on the amount of extra event relevant tweets (Rel) against the amount of extra event irrelevant tweets ($iRel$) in any of the adaptive dataset (AD), additional metrics are defined to measure their proportion to the volume of tweets in baseline dataset (BL) quantitatively. The amount of tweets in baseline is used as the reference to provide a parallel comparison across different KwAAs. In order to quantify the proportion of extra event relevant tweets, this research defined the **Information Gain** G by the equation 3.8. The information gain is negative when the volume of event relevant tweets in the adaptive dataset is less than that in the baseline one.

$$G = \frac{Rel_{AD} - Rel_{BL}}{Rel_{BL} + iRel_{BL}} \tag{3.8}$$

The other metric, measuring the proportion of event irrelevant tweets, is defined as **Noise Level,** N. Similarly, this is based on the amount of event irrelevant in an adaptive dataset ($iRel_{AD}$). The formula for this metric is calculated by equation 3.9.

$$N = \frac{iRel_{AD} - iRel_{BL}}{Rel_{BL} + iRel_{BL}} \tag{3.9}$$

Similar to the function of F-measure to Precision and Recall, it is important to formulate a single metric for assessing the performance of adaptive crawling under different events. As a result, the information gain G and the noise level N are combine together by

equation 3.10.

$$GNR = \frac{G}{N} = \frac{Rel_{AD} - Rel_{BL}}{iRel_{AD} - iRel_{BL}} \tag{3.10}$$

The information gain to noise level ratio, i.e. GNR, quantifies the ratio between the extra event information to that of extra event irrelevant information. When the adaptive crawler collects the same level of extra relevant tweets and irrelevant tweets, GNR will equal to 1. A GNR value less than 1 indicates that the extra noise is much larger than the extra event information, while a GNR larger than 1 means the adaptive crawler collects more event relevant information.

The adaptive crawling model contains scheme to guarantee that all the baseline tweets are collected by the adaptive crawling, unless the rate limits are hit. Consequently, by using the aforementioned grading strategy, the amount of event irrelevant tweets in the baseline dataset is always less than that in the adaptive datasets. Although the TF-KwAA reaches the rate limits during the crawling, the majority of traffic is noise as demonstrated in Figure 3.10 (b). As a result, the amount of event irrelevant tweets in TF-KwAA dataset is always larger than that in baseline dataset, i.e. N is always greater than zero. As the GNR metrics defines the way to quantify the extra tweets traffic introduced by the adaptive crawlers), the comparison between different algorithms becomes more straightforward. Figure 3.11 visualises the combined metric GNR based on the datasets retrieved by different adaptive crawlers during Glastonbury Festival.

The gain to noise ratio in Figure 3.11 are calculated on time frame basis. As can be seen from the figure, the information gain of TF-KwAA is always lower than 1. This indicates that the amount of extra noise in TF-KwAA dataset is always higher than the amount of information. This is also illustrated by Figure 3.10 where the volume of event-irrelevant tweets in TF-KwAA dataset is much higher than that of event-irrelevant tweets. The GNR is higher at the beginning of the evaluation period. Tweets about the performance in previous day, such as *Watching ArcticMonkeys at Glas they were fuckin awesome as always can't wait to go see them again :) #ArcticMonkeys*, result in the initial higher ratio. When approaching to the end of the evaluation period, the amount of extra noise in the TF-KwAA dataset can be 100000 times than that of extra information. The highest

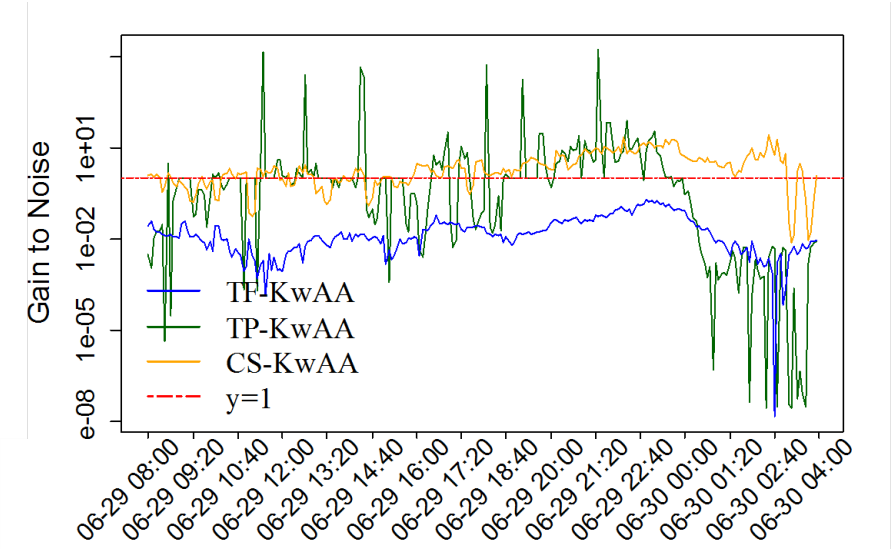


FIGURE 3.11: Information Gain to Noise Level Ratio for 2013 Glastonbury Festival (Evaluation Period)

GNR value is achieved during the **performance period** (from 20:00 to 00:00 when the volume of tweets in baseline is apparently higher, as shown in Figure 3.10), though still distance from the red line $GNR = 1$. Based on this observation, it is apparent that the TF-KwAA is collecting on random keywords. Without the constraint of rate limits, the TF-KwAA equipped adaptive crawler will collect on high frequency hashtags which mostly carry very limited extra event information. Similar to the TF-KwAA, both the TP-KwAA and the CS-KwAA achieve high GNR during the performance period. Though both algorithms show promising in identify more additional event relevant tweet, the pattern for this retrieving process is different. GNR varies for TF-KwAA during performance period while remain flat for CS-KwAA. In addition, as shown in Figure 3.11, a large amount of noise is introduced by TP-KwAA after the performance period due to the constant expansion of keyword list. This indicates that the CS-KwAA is more stable and constantly maintain higher GNR in the retrieval process. Namely, among these three KwAAs, CS-KwAA outperforms the TF-KwAA and TP-KwAA on extra event content identification.

3.6.1.4 Discussion

To sum up, this experiment demonstrates the proposed adaptive crawling introduces additional tweets in an automatic way during the real-time crawling. By conducting investigations on the proportion of keywords and tweets that are relevant to the event across datasets retrieved with all three KwAAs, this research then demonstrates that both the TP-KwAA and the CS-KwAA identify a notable amount of event-related keywords, which help to retrieve a greater amount of event relevant tweets. On the other hand, with additional keywords for crawling, the adaptive crawlers also collect some event irrelevant tweets. This is most obvious for the TF-KwAA. The statistics on the crawling results shows that datasets retrieved with TF-KwAA are easily polluted by noisy tweets. In addition, the amount of event-related tweets in TF-KwAA is less due to the rate limits. The other two KwAAs overcome this issue with additional filtering on keywords select, where the CS-KwAA outperforms the TP-KwAA by a constant performance without keywords outbreak. Although the TP-KwAA introduces extra event information during the event period, the later performance shows its venerability to the external changes. Once the TP-KwAA equipped crawler is exposed to the non-related keywords and event irrelevant tweets, it is unlikely to recover from the failure and thus lead to the outbreak of irrelevant tweets.

3.6.2 Comparison over Different Events

In addition to the evaluation of all the proposed KwAAs over a planned event, their performance under additional events is also investigated. As demonstrated in section 3.6.1, TF-KwAA introduces too much noise to be considered a viable algorithm, it will no longer be discussed in this sub-section. Namely, this sub-section explains and illustrates the result on datasets crawled by TP-KwAA and CS-KwAA, covering both planned and unplanned events. A similar evaluation process used in the previous experiment is inherited. This subsection first introduces the event datasets for this cross-event evaluation (in section 3.6.2.1). Then, the performance results on the identified keywords (in section 3.6.2.2) and retrieved tweets (in section 3.6.2.3) are given. Finally, this subsection

concludes this evaluation with a discussion of on two hypotheses: the proposed adaptive crawling gain extra event relevant information in diverse scenarios and it constantly acquire additional event relevant data for both planned and unplanned events (in section 3.6.2.4).

3.6.2.1 Event Datasets

An event, classified by its type, falls into either planned or unplanned event [70]. The event used in the previous experiment (as described in section 3.6.1) is a planned event as the information (e.g. time, address, content) is known before the event happens. It has been demonstrated that the adaptive crawlers introduces both additional event relevant information and event irrelevant noise. A further evaluation is necessary to prove the proposed adaptive crawling model can achieve reliable performance over multiple events and different type of events. In this evaluation, datasets retrieved by TF-KwAA are not considered since it is likely to be polluted by noises. The TP-KwAA is retained for analysis because it provides event information during performance period. This further experiment is carried on datasets crawled by different KwAAs during three additional events, including one additional planned event (i.e. 2013 Wimbledon Championships[9]) and two unplanned event (i.e. Egypt Protests[10] and Malaysia Airlines Flight 370 (MH370) Missing Plane[11]). Table 3.9 presents the overview of all the evaluated events.

In order to prevent the bias toward a good result, the initial keywords used for all the events are general terms, and usually in plain text (without # symbol). According to the keyword matching rule by Twitter (as described in section 2.3.1.3), when querying the core database with term X in plain text, tweets containing X and tweets containing its hashtag format $\#X$ will be retrieved. If only hashtags are used as the initial keywords, the baseline dataset would be significantly smaller. On average, more than 50% of tweets contain hashtag in the baseline event datasets (in Table A).

[9] 2013 Wimbledon Championships: `http://www.wimbledon.com/`
[10] Egypt Protests: `https://en.wikipedia.org/wiki/2012%E2%80%9313_Egyptian_protests`
[11] MH370 Missing Plane: `https://en.wikipedia.org/wiki/Malaysia_Airlines_Flight_370`

TABLE 3.9: Event datasets overview for Evaluation and Comparison

Planned	Glastonbury Festival			Wimbledon Championship		
	Baseline	TP-KwAA	CS-KwAA	Baseline	TP-KwAA	CS-KwAA
Init. Keys	Glastonbury			#wimbledon2013, Wimbledon		
Period	2013-06-29, 08:00 to 2013-06-30, 04:00 (20h)			2013-06-26, 00:00 to 2013-06-27, 00:00 (24h)		
Tweets No. (ave. rate/min)	206,559 (172)	559,663 (466)	287,254 (239)	429,699 (198)	1,119,178 (777)	483,624 (336)
Keys No. (ave. rate/hour)	1 (-)	917 (46)	194 (10)	2 (-)	1,772 (74)	307 (13)

Unplanned	Egypt Protest			MH370		
	Baseline	TP-KwAA	CS-KwAA	Baseline	TP-KwAA	CS-KwAA
Init. Keys	Egypt protest, #ArabSpring			MH370, Malaysia Airlines		
Period	2013-07-17, 21:20 to 2013-07-18, 16:40 (19.3h)			2014-03-20, 11:20 to 2014-03-20, 14:10 (2.8h)		
Tweets No. (ave. rate/min)	76,993 (66)	716,939 (619)	219,152 (190)	68,986 (406)	342,886 (2019)	73,883 (435)
Keys No. (ave. rate/hour)	2 (-)	1,156 (60)	211 (11)	2 (-)	807 (285)	43 (15)

The parameters follows the same setting as that used in the previous experiment (as tuned in section 3.5), while the period of evaluation is chosen based on the characteristic of different events. For the planned event, i.e. Glastonbury and Wimbledon, the goal is to select a period that the tweets traffic vary significantly enough to cover both peaks and valleys. The selection of evaluation period for unplanned event is more restricted. It is impossible to anticipate when the event will happen and when to initiate the crawlers for retrieving event content. The generic periods during the aftermath of the events are selected. Although the evaluation period for MH370 is shorter due to the down time of API connection [12], adaptive crawlers with both KwAAs show similar performance during other crawling periods. The length of evaluation period varies from event to event, ranging from the shortest 2.8 hours to 24 hours.

According to Table 3.9, it is obvious that the TP-KwAA equipped crawler identifies the most keywords and introduced the most tweets traffic for all the events in this

[12]The down time is from 14:20 to 17:20

experiment. Rather than comparing the volume of keywords and tweets directly, their arrival rate during the evaluation periods are calculated. According to the figures in the table, MH370 results in the highest arrival rate for all three crawlers. The datasets crawled by TP-KwAA equipped crawler achieves the highest arrival rate (285 keywords per hour and 2019 tweets per minute). On the other hand, the second place is hit by the 2013 Wimbledon Championship, rather than the other unplanned event. The arrival rate is 74 and 13 keywords per hour, 777 and 336 tweets per minute for TP-KwAA and CS-KwAA respectively. The difference of planned event and unplanned event, in regarding to their arrival rate, is not obvious according to this table. This is caused by the fact that the tweets traffic volume and pattern of different events is associated with how the events attracts people's attention. Therefore, the figures are event dependent. The following experiments investigate the performance and pattern of different KwAAs during the four different types of events.

3.6.2.2 Event-relevance of Identified keywords

By using the same hashtag categorization strategy, this research investigates the keyword composition for all the event datasets listed in Table 3.9. The results of keywords categorization is illustrated by Figure 3.12. Here, a bar plot is presented in order to illustrated the keyword composition of the two different KwAAs across all four events. The keyword composition for the baseline crawler is fixed to the seed terms and thus is 100% C1. It is not presented in the figure. Each bar in the figure represents the composition of all the distinct keywords of a single event. The eight bars are grouped based on the different KwAAs listed at the top. The left four bars show the keywords identified by TP-KwAA for the four events, while the right illustrates the composition of keywords which are identified by CS-KwAA. The total keyword number and the amount of keywords in each category, during the whole evaluation period, can be observed from y-axis. Namely, the number of related (C1), possibly-related (C2), not known (C3) and non-related (C5) and their percentage to the total number of keywords is accumulatively shown in the bar. Since this section concerns the relevance of identified keywords, the statistics of non-keyword (C4) is not presented.

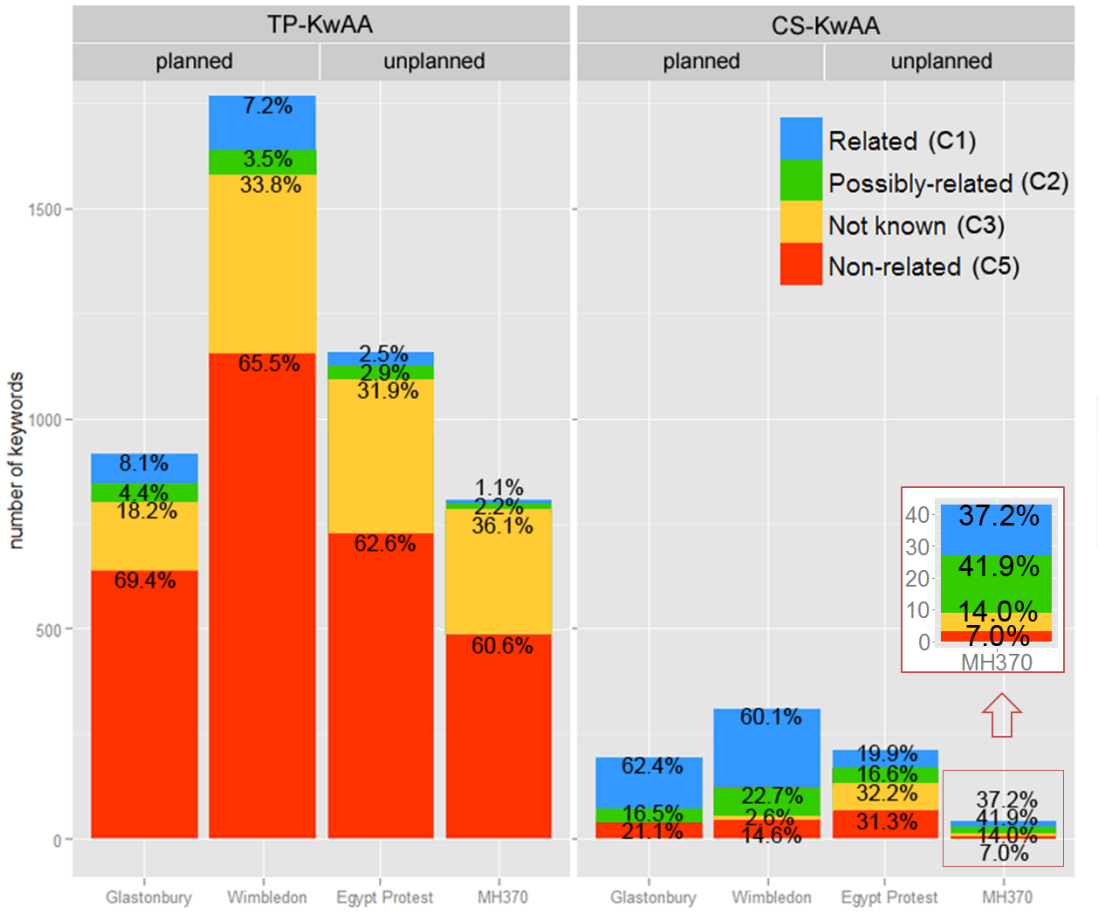


FIGURE 3.12: Keywords Categories and Distribution of Four Evaluation Datasets

As can be observed from Figure 3.12 the proportion of "not known" C3 keywords are higher for the unplanned event. This is especially significant for the CS-KwAA, where the proportion difference can be as high as 32.2% (32.2% in Egypt Protest keywords, while no C3 keyword for Glastonbury). In fact, the usage of hashtags for unplanned event is more ambiguous. There exists two reasons for the higher proportion of C3 keywords. Firstly, people tend to use some general hashtags collaboratively with event specific hashtags when post tweet during unplanned events. These general terms can be difficult to categorize and thus been assigned to C3. Secondly, during the keyword labelling process, the ambiguous use of hashtags results in the higher chance of disagreement on keyword categorization. Therefore, the number of C3 keywords increase. Also, the proportion of C1 and C2 keywords for planned event is much higher than that

of unplanned events, as can be concluded from Figure 3.12. If only considering the proportion of related keywords, both algorithms achieve the best performance during the 2013 Glastonbury Music Festival, while the show the worst performance during unplanned events (MH370 for the TP-KwAA, and Egypt Protest for the CS-KwAA).

Figure 3.12 illustrates how CS-KwAA achieve a better performance than the TP-KwAA in keyword identification under both planned and unplanned events. It is obvious that CS-KwAA is more stable by observing the proportion of different classes of keywords. The average percentage of C5 keywords by TP-KwAA is almost three times of that by CS-KwAA. This huge difference indicates that the amount of event irrelevant tweets in TP-KwAA datasets is large. Furthermore, the differences between C1 and C2 keywords for these two algorithms further demonstrates the previous observation: CS-KwAA is better than TP-KwAA in event-related keyword identification. Even though the total number of keywords in CS-KwAA is always far less than that of TP-KwAA, there are more C1 and C2 keywords in CS-KwAA than TP-KwAA, no matter what event is. Specifically, the C1 and C2 keywords only take up to 12.5% of the keywords list when using TP-KwAA, while the proportion of C1 and C2 keywords is always higher than 50% for the CS-KwAA group. The only exception is Egypt Protest event. There are two reasons for this exception. First, some event-related keywords can be very hard to assign to correct category. Even though there is participate labelling the hashtag to the right category, it can be considered as C3 at last due to the higher disagreement for keywords of unplanned event. Another reason is more event specific. The Egypt protest is an activity popular among non-English speaking country with some amount of hashtags (20%) and tweets written in Arabic. As many keywords are also in Arabic, the labellers find it more difficult to categorise them even with the help of translation tool. Figure 3.12 shows that many more keywords (more than 30%) in Egypt Protest are labelled with C3.

3.6.2.3 Event-relevance of Retrieved Tweets

This research then investigates the relevance between the additional tweets and the events through an experiment similar to that given in 3.6.1. All the retrieved tweets

are classified into either event relevant or event irrelevant by using the grading strategy from Table 3.1. Here, the results are illustrated by the more intuitive information gain and noise level. Figure 3.13 plots the information gain to noise level ratio across all the evaluation events.

Figure 3.13 illustrates the GNR of four different events, where (a) and (b) are the results for planned event and (c) and (d) are that for unplanned event. As shown in the figures, the chance for the amount of extra event relevant tweets higher than irrelevant tweets is always larger for the planned events. This is more obvious for the CS-KwAA datasets. In this experiment, the observation is similarly to what has been concluded in the keyword composition experiment: both KwAAs perform better during the 2013 Glastonbury Music Festival than the other planned event. Besides, the plots show that the CS-KwAA constantly gets higher GNR during the event period for both planned event (around 18:00 to midnight for Glastonbury Festival and 12:00 to 17:00 for Wimbledon Championship). On the other hand, the performance of these automatic adaptive crawlers is less optimal during unplanned event. As shown in Figure 3.13 (d), the TP-KwAA equipped crawler gets more extra irrelevant tweets than relevant tweet almost during the whole evaluation period. It achieves a better performs during the Egypt Protest event. Though there exist multiple periods with bad GNR (less than one), the rest half of the evaluation period ended with good GNR (larger than one) by the TP-KwAA equipped crawler. However, the situation for the CS-KwAA is opposite. Although the performance of CS-KwAA also degrades for the unplanned events, results shows that it can still identify greater amount of extra relevant information than the irrelevant information at most of the evaluation period.

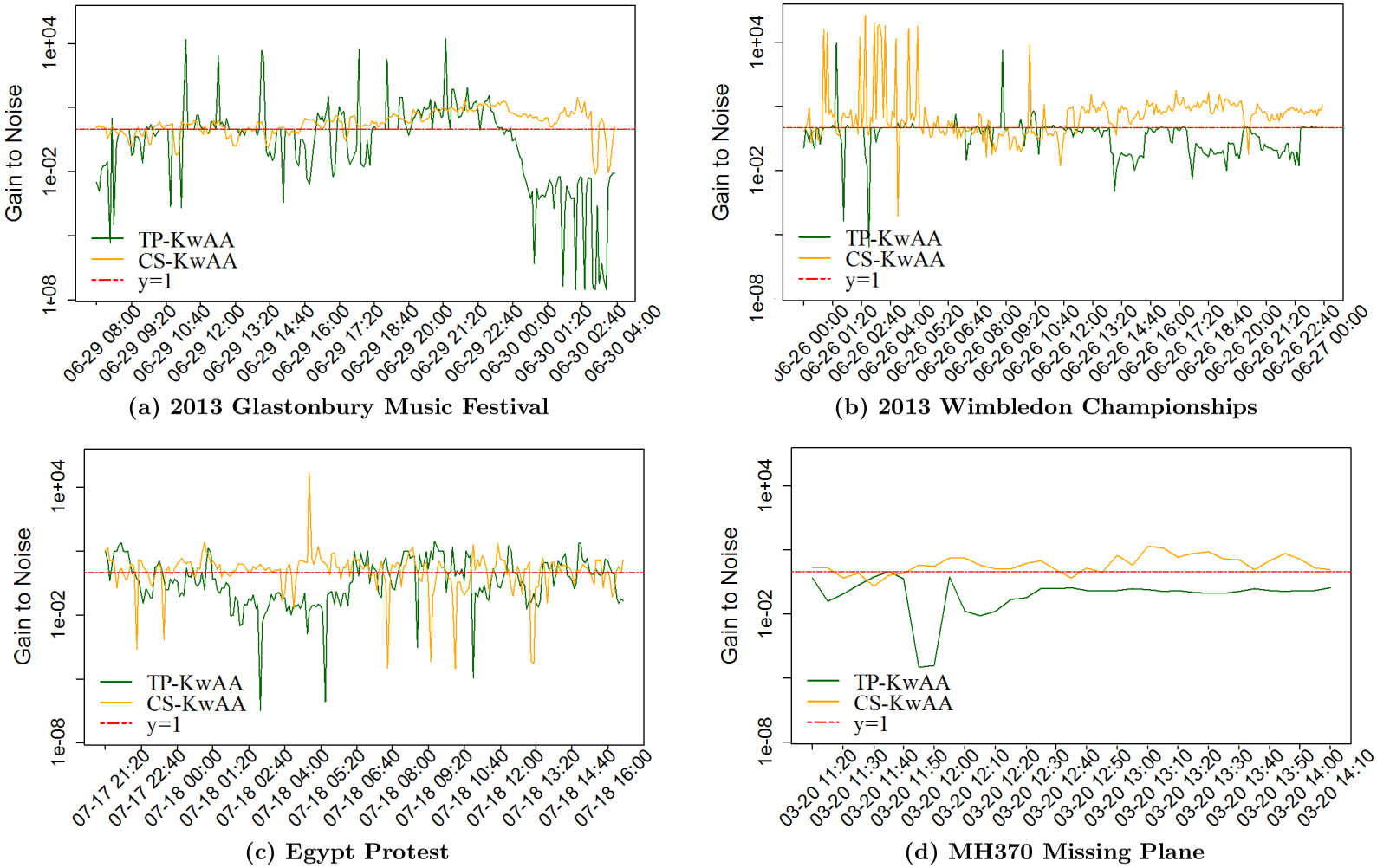


FIGURE 3.13: Information Gain to Noise Level Ratio of all the Evaluation Datasets (a) - (b) planned events; (c) - (d) unplanned events

3.6.2.4 Discussion

The findings about the KwAAs found by the preliminary experiment in section 3.6.1 are further supported by the experiment in this subsection. The findings here further demonstrate that the CS-KwAA is more reliable than the TP-KwAA. Sudden outbreaks of irrelevant tweets are less likely happen to CS-KwAA. No matter what type of events and what stage of the event, CS-KwAA achieves good perform once as shown by its usually good GNR. Even if it suffers from the sudden outbreak of irrelevant tweets (as shown in Figure 3.13 (c) when the plot for CS-KwAA below $y = 1$), CS-KwAA drops the non-related event keywords quicker than TP-KwAA. In addition, this experiment also reveals characteristics of the proposed adaptive crawling and KwAAs during both type of events. The proposed KwAAs achieve better performance under planned events in terms of both keyword identification and the gain to noise ratio. Their benefits on extra event content identification is more visible during the event period. As for the unplanned events, the CS-KwAA shows much better performance than the TP-KwAA. As concluded previously, the TP-KwAA is more vulnerable to noise during unplanned events. The amount of related keywords is often very small for TP-KwAA and thus the GNR is lower than one.

3.6.3 Performance Discussions

Based on the observations and analysis of above experiments, this research conclude several **properties** for the proposed adaptive crawling model, as well as the proposed KwAAs. This section summarises these properties with a detailed discussion.

P1: *TP-KwAA is a language independent algorithm that can identify both English keywords and keywords in other language.*

According to the keywords list generated by different KwAAs, it is noticed that the TP-KwAA is more likely to include keywords in languages other than English. When the TP-KwAA behave normally, i.e. not hits the Twitter rate limits, the non-English keywords are always relating to the event. In terms of developing algorithm to collect the largest amount of event content, this property is very desirable. Both English hash-

tags and non-English hashtags are represented by a series of count, so their relevance to the event is only calculated by the similarity between count series. CS-KwAA is unable to achieve this as it favours the hashtags written in the same language as the initial keywords. For example, the keywords used in this research are all in English, so the retrieved tweets and hashtags in those tweets are likely written in English. Even though hashtags exist in other languages, they tend to be eliminated when comparing the content similarity as tweets in English can't be similar to non-English tweets. As result, identifying new keywords that are not in English is harder when the tweets content is the only reference.

P2: *CS-KwAA performs the best among all three KwAAs for topic keyword identification and event tweets retrieval.*

As show in the previous section 3.6.1 and section 3.6.2, the performance of CS-KwAA is the most stable regardless of the event types. The lack of criteria to filter out event irrelevant keywords makes the TF-KwAA very vulnerable to noise. The top ranked hashtags and Twitter's trends[13] are not always about the real-world event. This further reduces the stability of TF-KwAA while crawling. On the other hand, TP-KwAA can be unstable when shifting from different phases of event or types of event. It has been shown that the hashtag frequency returned back by Streaming API is often misleading when capped with rate limits, especially for the top ones [80]. This indicates that the frequency value from the streaming API is not always reliable as it sometimes doesn't consistently gives the right count and proper tweets. Since both TP-KwAA and TF-KwAA rely on the frequency of hashtags, the performance of them depends on the status of Twitter Streaming API. These two algorithms thus suffer from outbreaks when they approaches the rate limits (as shown in Figure 3.9 after 1:00). By learning the lesson from the previous two KwAAs, this research then proposes the CS-KwAA to overcome the issues in other KwAAs. Results show that this new KwAAs not only identify larger amount of event keywords and tweets, but also keep away from being polluted by event irrelevant information.

P3: *Planned events are more apt for adaptive crawling and our proposed KwAAs.*

[13]Twitter provides lists of trending topics worldwide or within a specific area. These lists contain terms that are mentioned at a greater rate than others

Another observation from the above experiments is that the KwAAs seem to get better performance for planned event. In fact, this is in common with other research [49] as the characteristics of planned events are easier to estimate and prepared. Although the unplanned events are harder to estimate, the labelling process is one of the main reasons that lead to the less satisfactory results for unplanned events. The lack of structured materials of unplanned makes the labelling process harder. Hashtags which are generated during those unplanned events are more ambiguous to assign with a category. Also the unplanned events used in our evaluation contains more foreign keywords that made the labelling process complicated. Some of the C3 non-English hashtags can be relevant to the event. When they are not assigned to the correct categories, the results become less satisfactory.

3.7 Summary

This chapter presents an automatic event content collection method which is capable to gather a set of tweets, without preliminary knowledge of the events, by just relying on initial search terms for live events. To be more specific, the proposed adaptive crawling model allows comprehensive information about an event to be retrieved. By equipping the Keyword Adaptation Algorithm, the proposed adaptive crawling model can collect an extended set of specific instances of an event. This is achieved by monitoring the Twitter live stream with only the initial keywords, without manual modification of the search terms. In designing the adaptive crawling model, the challenge is to identify extra search terms, beyond the original keywords, appearing in content related to the event in question. Specifically, the key aspects of the work in this chapter are as follows:

- an modification of the traditional Twitter crawling model to allow automatic and real-time adaptation by incorporating an adaptive crawling module
- an investigation on different ways of introducing additional Twitter traffic to the proposed adaptive crawling model, including the use of Term Frequency, Traffic Pattern and Content Similarity with the KwAA

- an examination on the proposed adaptive crawling model and three KwAAs by applying them to four different events that belongs to two generic types of real-world events.
  - the first experiment shows that the content similarity based KwAA performs better than the others in terms of topical keyword identification and event relevant tweets retrieval.
  - the second experiment shows that the proposed adaptive crawling gain extra-relevant information not only in diverse events scenarios but also for both planned and unplanned events.

To sum up, by investigating the crawling results of the proposed adaptive crawling and all three KwAAs with multiple real-world events, this chapter demonstrates that the adaptive crawling introduces additional topical keywords and event content during automatically crawling. Compared with the unplanned event, the adaptive crawling model is apt to the planned events. In addition, the comparison of different KwAAs over multiple events also reveals that the adaptive crawling equipped with CS-KwAA performs best on additional event-information identification, for both new keywords and tweets. It is more stable and get constants good performance regardless of the stage and the type of events.

Chapter 4

Event Detection with Adaptive Microblog Crawling

Through the experiments over event datasets with different characteristics (such as different event types, volume of traffic and durations), chapter 3 demonstrated that an extra amount of event content can be retrieved from Twitter in a real-time manner. In this chapter, the aim is to investigate the feasibility of using this extra event content in the real-time event detection task. Most of the existing research in event detection focuses on improving the accuracy of Twitter event detection via the modification of TDT algorithms, this research explores the benefits of the extra event relevant content collected by the proposed adaptive crawler. The questions of interests in this chapter are:

1. Whether the additional event content retrieved by the adaptive crawler helps to improve the accuracy of sub-event detection?
2. Does these additional content contribute to better event awareness in terms of the number of sub-events can be detected and the amount of information conveyed by them?

This chapter presents a Twitter event monitoring solution, called "Detection of Sub-events by Twitter Real-time Monitoring (DSTReaM)". This TEM framework provides

a mean to quantitatively analyse the effects of the additional event content to the event detection results. In the scenario of this research, the stream analysed is associated with a specific event. Therefore, the DSTReaM employs a statistical-based outlier detection algorithm that relies on the temporal feature of the event stream. The assumption made by this kind of algorithms is that an event is always accompanied by the sudden raise of people's interests, and therefore the volume of tweets talking about it is also increasing.Specifically, in order to exploit the usefulness of extra event content in the sub-event detection, the proposed framework decomposes the original event stream into multiple threads and applies the peak detection algorithm parallel to each sub-stream.

The structure of this chapter is as follows: Firstly, section 4.1 gives an introduction on the DSTReaM. In section 4.2, the preparation work and methodology of the investigation is addressed. This includes the required pre-processing of the original datasets, parameters tuning based on the previous analysis and evaluation metrics definition. Section 4.3 investigates the performance of the DSTReaM with two planned events (Glastonbury Music Festival and 2014 Sochi Olympic Games). Finally, a summary is provided to highlight important observations of the two experiments in section 4.4. It also highlights other potential way of integrating adaptive crawling and event detection algorithm.

4.1 Detection of Sub-events by Twitter Real-time Monitoring (DSTReaM)

In order to detect and summarise the sub-events that happen during the user-specific news event, this research proposes an event monitoring framework, i.e., DSTReaM, that follows the content retrieval, detection and summarisation pipeline presented in section 2.4.1. This is achieved by building upon the existing event detection algorithm Twitinfo [43]. By collecting, detecting and then extracting the most descriptive event-relevant tweets, the framework automatically formulates descriptions of the sub-events. Whilst existing research focuses on the depth of detection, i.e., on achieving more accurate detection results, the focus of this research is to achieve higher accuracy on sub-event detection by increasing the coverage of the event content, i.e., number of plausible dis-

tinct events.

As shown in Figure 4.1, the proposed event monitoring framework contains three main components. The **Adaptive crawler** component (as introduced in chapter 3) collects

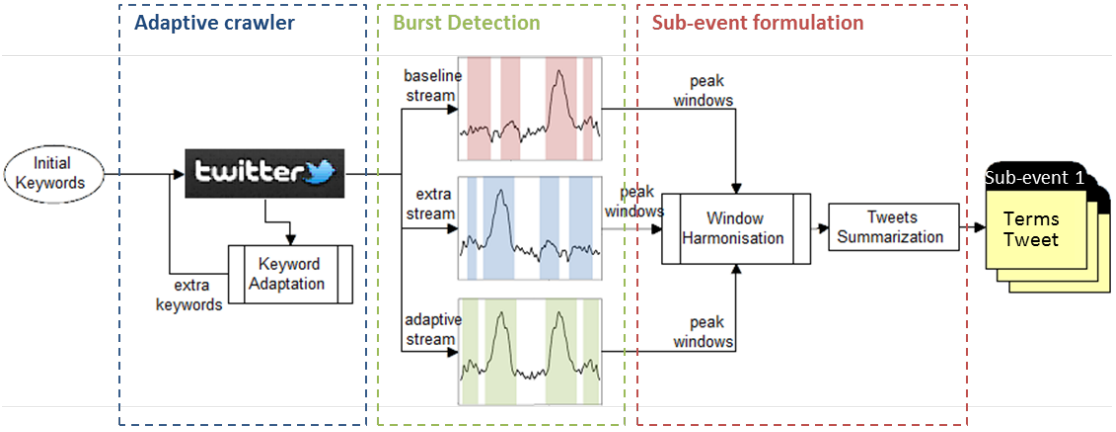


FIGURE 4.1: Twitter Event Monitoring Solution: DSTReaM

a comprehensive set of event tweets. It produces a stream of event tweets analysed by the **Parallel Bursts detection** component. The bursts detection component identifies the potential sub-events. Each potential sub-event is represented by a timespan and a collection of tweets whose timestamps fall within the timespan. After the detection of peak window, all the potential sub-events are post processed by the **Sub-event formulation** component to finalise the description of the detected sub-events. This framework is initiated by a set of user-specified keywords that target the crawler on a particular event. The output is a list of sub-events which are specified by a timespan, a group of descriptive terms and a summary tweet.

Detecting events by the statistical count of tweets features is a widely used approach in Twitter events detection. This approach is based on a common observation that a burst of features is positively correlated with the occurrence of a real-world trending event. Therefore, the real-world events can be identified by capturing the sudden bursts of tweet features. As described in the previous section 2.4.3, the features used for mining such relationships are very diverse: the volume of individual terms can be used as separate features while sometimes the whole Twitter stream is considered as a single feature. This research is interested specifically on the stream about a specific event, referred to

event stream. In the DSTReaM framework, the event stream is retrieved by either a baseline crawler or the adaptive crawler with CS-KwAA.

The rationale to select the volume of whole event stream, rather than the volume of terms or phrases, as the detection unit is threefold. First, the proposed KwAAs introduce biases when detect events by bursting terms. When the adaptive crawler collecting new tweets with the identified keywords, the terms, which are the time frame keywords, are more likely to show the bursting behaviour. Second, the detection of event in finer granularity level (e.g. n-gram, named entity or other tweets units) always requires additional resources and processing power for segmenting the tweets and thus cannot scale to handle the massive volume of Twitter streams. Last, segmenting tweets ruins the semantic coherence of the sentences and results in the fragmented detection results.

4.1.1 Adaptive Crawler

As shown in chapter 3, the Adaptive Microblog Crawling model improves the comprehensiveness of event content coverage. As a result, this framework uses the adaptive crawler for identifying event topics that arise in the midst of events and the expanded set of event information. The CS-KwAA provides the most reliable results among the three algorithms and is used to provide an expanded event stream for the rest of the framework (as demonstrated in section 3.3).

4.1.2 Parallel Burst Detection

In this component, an existing burst detection algorithm used by Twitinfo [43] is parallelly distributed to a multi-threaded event stream for identifying different layers of potential sub-events. The original Twitinfo algorithm is inspired by the conventional statistical model based outlier detection algorithms but improves the detection performance by using a smoothing technique. Specifically, the author of Twitinfo system exploits the exponentially weighted moving average (EWMA) in TCP congestion control [126] to identify the relatively maxima, i.e. burst, by considering the recent history.

Although the EWMA is designed to find the outlier of package arriving rate in a communication channel, it can be borrowed in the event detection task for distinguish the local extreme value for the tweets' arrival rate. In general, this algorithm starts a peak window for an event when it encounters a significant increase in tweet volume. The end of the peak window is identified when the tweet volume returns to the same level as when the burst started, or a new peak window is identified. Consequently, a peak window is a pair of timestamps, where the first timestamp defines the moment when the detected event starts and the other one defines the moment when that ends. The detailed pseudo code is listed in Algorithm 5.

The TwitInfo algorithm first calculates the tweets arrival rate based on the bin defined by the length of time slots. This length is based on the characteristics of the event and can be manually determined before the detection. As a result, the current tweets arrival rate C_i thus can be calculated based on the pre-defined bin size. After that, the system applies the EWMA mechanism for the expected arrival rate. This is achieved by estimating the mean average μ as well as the mean deviation $\sigma_{(\bar{C})}$ of the history arrival rates, as show in equation 4.1 and 4.2 respectively.

$$\mu_i = \alpha \cdot C_i + (1 - \alpha) \cdot \mu_{i-1} \tag{4.1}$$

$$\sigma_{(\bar{C})_i} = \alpha \cdot |\mu(i) - C_i| + (1 - \alpha) \cdot \sigma_{(\bar{C})_{i-1}} \tag{4.2}$$

If the arrival rate of tweets in a time slot is significantly higher than the expected tweets arrival rate (as calculated by the inequality equation 4.3), the time slot of this local maximum and other slots around it are labelled as a peak window, and thus been interpreted as an event.

$$C_i \geqslant \mu_{i-1} + [\tau \cdot \sigma_{(\bar{C})_{i-1}}] \tag{4.3}$$

To maximise the utilisation of the extra event content identified by the adaptive crawler, this peak detection algorithm is not only applied to the adaptive stream, but also to its decomposed streams. Specifically, three instances of the Twitinfo event detection algorithm are run in parallel over the three decomposed stream, i.e. the baseline stream,

Algorithm 5 Twitinfo Event Detection Algorithm

Require: Tweet arrival counts $C_{all} = \{C_1, C_2, ...C_n\}$,
Detection latency p, Smoothing factor α, Inequality threshold τ

function find_peak_window (c, p, α, τ)
    windows $= \varnothing$
    $\mu = C_1$
    $\sigma_{(\bar{C})} = \text{MAD}^{\dagger}(C_1, ..., C_p)$

for $i = 2; i < length(C_{all}); i++$ **do**
  if $C_i \geqslant \mu_{i-1} + [\tau \cdot \sigma_{(\bar{C})_{i-1}}]$ **then**
    start = i-1
    while $i < length(C)$ and $C_i > C_{i-1}$ **do**
      $(\mu_i, \sigma_{(\bar{C})_i}) = \text{update}(\mu_{i-1}, \sigma_{(\bar{C})_{i-1}}, C_i)$
      i++
    end while
    while $i < length(C)$ and $C_i > C_{start}$ **do**
      if $C_i \geqslant \mu_{i-1} + [\tau \cdot \sigma_{(\bar{C})_{i-1}}]$ **then**
        end = –i
        break
      else
        $(\mu_i, \sigma_{(\bar{C})_i}) = \text{update}(\mu_{i-1}, \sigma_{(\bar{C})_{i-1}}, C_i)$
        i++
      end if
    end while
    if $C_i < C_{start}$ **then**
      end = i++
    end if
    windows = windows ∪ (start, end)
  else
    $(\mu_i, \sigma_{(\bar{C})_i}) = \text{update}(\mu_{i-1}, \sigma_{(\bar{C})_{i-1}}, C_i)$
  end if
end for
return windows
End function

function update $(\mu_{i-1}, \sigma_{(\bar{C})_{i-1}}, C_i)$
    $\mu_i = \alpha \cdot C_i + (1 - \alpha) \cdot \mu_{i-1}$
    $\sigma_{(\bar{C})_i} = \alpha \cdot |\mu(i) - C_i| + (1 - \alpha) \cdot \sigma_{(\bar{C})_{i-1}}$
return $(\mu_i, \sigma_{(\bar{C})_i})$
End function

†MAD: Mean Absolute Deviation

adaptive stream and the extra stream, as illustrated in Figure 4.1. Here, the baseline stream is made of event content identified by the baseline crawler with the same initial keyword as the adaptive crawler. The extra stream is composed by the tweets that can be identified by the adaptive crawler but not the baseline crawler, i.e. the stream

obtained based on the equation 4.4. For any of the decomposed streams, a list of peak windows is generated and sent to the next step.

$$\text{extra stream} = \text{adaptive stream} - (\text{adaptive stream} \cap \text{baseline stream}) \tag{4.4}$$

4.1.3 Sub-event Formulation

The lists of peak windows, i.e. output from the parallel burst detection component, don't provide the context information about the sub-events, they only demonstrate the timestamps when there is abnormal burst on tweets volume. As a result, this research extracts the textual information from the decomposed sub-streams for describing all the detected peak windows. The number of tweets retrieved for any peak window is still very large and thus representative tweets from this window are chosen. The sub-event formulation process consists of two sub-steps:

a. Window Harmonisation Each peak window W that is detected from one of the Twitter stream among BL, AD and EX can be described by the most frequent unigram, measured by their TF-IDF value. The assumption here is that the summary term of one window should be very different from other windows. Therefore, the TF value is calculated by all the tweets in window W, and the IDF is based on the tweets in all the previous identified peak windows. Following the same strategy as in the Twitinfo system, the top 5 TF-IDF weighted terms with highest TF-IDF value from each peak window are selected to represent the corresponding potential sub-event. Similar to the Twitinfo system, the number of summary terms is set to 5.

As shown in Figure 4.1, the burst detection algorithm is applied in parallel to all three streams, i.e. AD, BL and EX. Consequently, the detected bursts can be represented as a list of peak windows and their corresponding terms, i.e. summary terms. These terms are ranked based on their TF-IDF value. However, when running the burst detection algorithm over multiple streams, there is probability that peak windows from different event streams represent the same sub-event. In the proposed framework, a

window combination step is employed to reduce the amount of duplicated peak windows. In this sub-step, two peak windows that are detected from different streams are considered as duplicated if the peak windows overlap in time and contain more than half common summary terms. Once the two windows are recognized as duplicated, they are combined together and considered as single peak window. We use the same properties, i.e. timespan and summary terms to describe the combined window. The new timespan is calculated as the union of the individual timespan of all the duplicated windows, while the summary terms are recalculated based on the same strategy for the summary selection process which is used in the Twitinfo system.

b. Summary tweets This research takes advantage of tweets in the peak windows for a structured summarisation. In this approach, the score of a tweet is the average TF-IDF value of all the terms that appear. This final score, i.e. also known as "normalized TF-IDF score $score(\overline{TF-IDF})$", is calculated by equation 4.5.

$$\overline{TF-IDF} = \frac{1}{n} \cdot \sum_{i=1}^{n} tfidf_i \tag{4.5}$$

Before the calculation, all the tweets are pre-processed with stop word and punctuation removal, stemming and Twitter symbols (@ user mention and shorten URL) filtering, where the remaining terms are referred as *informative terms*. However, the drawback of using the normalized TF-IDF score is that this strategy favours selection of short tweets that only contains few terms with very high TF-IDF value. Normally, the number of the distinct informative terms in the summary tweet is often less than 2, being primarily made up of terms with high TF-IDF values. Since these short length tweets typically don't provide any extra information over the summary terms, this research preferentially selects tweets with more different terms. Specifically, the summary tweets is expected to have the highest normalized TF-IDF score among all the tweets belonging to that peak window and have at least two terms. A longer tweet with the same normalized TF-IDF score as a shorter tweet will always be preferred.

4.2 Datasets Preparation and Investigation Approaches

In order to apply the Twitinfo event detection algorithm to the datasets retrieved by the proposed adaptive crawler, it is necessary to prepare the datasets, tune the algorithm parameters and define the evaluation metrics.

In this section, the overview of the tested datasets is addressed first (in section 4.2.1). Then the preparation work to both the datasets and algorithm is introduced, including the preparation of raw datasets to the algorithm (in section 4.2.2) and the tuning of algorithm parameters (in section 4.2.3). At the end of this section, the metrics for measuring and comparing the detection results are detailed (in section 4.2.4).

4.2.1 Event Datasets

To explore the detection benefits under different event scenario, this research select two events for the investigation: the *2013 Glastonbury Music Festival (Glastonbury Festival)* and the *2014 Sochi Olympic Games (Sochi Olympic)*. The timeline of the Glastonbury festival event is more intense because multiple performances are carried out simultaneously. On the other hand, the schedule of the Sochi Olympic event is sequential and even irregular because the time duration of each competition varies. A detailed overview of the evaluated datasets is shown in Table 4.1.

Table 4.1: Event Datasets Overview

	Glastonbury Festival		Sochi Olympic	
	Baseline	Adaptive	Baseline	Adaptive
Init. Keys	Glastonbury		Sochi,#olympic2014, #sochi2014	
Period (duration)	2013-06-29, 11:00 to 2013-06-30, 00:00 (4 hours)		2014-02-22, 05:15 to 2014-02-22, 19:15 (14 hours)	
Tweets No. (ave. rate/min)	171,254 465	232,811 645	213,986 255	281,692 335
Keyword No. (ave. rate/hour)	- -	118 29	- -	247 18

Following the same strategy used in the previous crawling, both events are crawled with plain text keyword ("**Glastonbury**" for 2013 Glastonbury Festival and "**Sochi**" for

2014 Sochi Olympic Games) and for Sochi Olympic, two specialised event hashtags are also employed. As mentioned in section 1.2, one of the research aims is to understand the evolution of an event stream by its sub-events. This research investigates the performance of the algorithm when the sub-events are reported by the news media. The tweets arrival rate of Glastonbury Festival is higher than that of Sochi Olympic, for both the baseline datasets and the CS-KwAA adaptive datasets. This is due to the higher keyword identification rate, i.e. more hashtags are used during the Glastonbury Festival if they all relate to the event. The following processing and evaluation are carried on these two datasets.

4.2.2 Datasets preparation for Event Detection

The online content generated by the general public, such as tweets, can be extremely noisy and unstructured. Although the baseline and the adaptive crawling model filtering the stream with a set of event specific keywords, the event stream retrieved by the baseline crawler and adaptive crawler still contains event-irrelevant information (as demonstrated in section 3.6.1). Detection of events over such data directly can lead to unexpected results. To better understand the effect of both event relevant and irrelevant content in the extra tweets, this research investigates the event detection results with both raw and filtered datasets.

- **Raw data from the crawlers (Raw)**

  The first series of experiments are carried on the raw data which is crawled by the baseline or the adaptive crawler. In this setting, the output of the crawler is sent to the detection algorithm directly without any additional processing. To simulate a real-time event detection scenario, all tweets are provided to the system in a streaming manner and processed in a single pass (no re-examination of the tweet once it is processed).

- **Dataset with only Event relevant tweets (Filtered)**

  In the second group of experiments, only the event-relevant tweets, as classified by the method in section 3.4.3, are sent to the detection algorithm as the input.

Although, additional data processing is required in this setting, this extra step reduce the impact of noisy tweets introduced by the adaptive crawling. Moreover, it is still possible to achieve real-time detection if an automatic tweet classifier is trained. The classifier don't need to be very accurate, the requirement is to filter the background noise, which can be achieved advance in offline [34, 127].

With the aim of a better understanding of how the additional content benefit the event detection, the investigation is conducted over three different Twitter stream for each event:

- **BL**: the common baseline which apply the detection algorithm directly to the pre-defined keyword specified baseline stream.
- **AD**: using the same detection algorithm as the BL approach but exploiting the adaptive stream that retrieved by the adaptive crawler with the same initial keywords as BL.
- **EX**: using the same detection algorithm as the BL approach but exploiting the extra stream that obtained by Equation 4.4.
- **ALL**: the proposed event monitoring solution that described in section 4.1.

The tweets volume of all the datasets used in the detection benefit investigation are listed in Table 4.2. "Proportion" is the ratio between the number of tweet in Filtered datasets and that in Raw datasets, i.e. the proportion of event-relevant tweets in the event stream.

TABLE 4.2: Tweets Volume of Evaluation Datasets

	Glastonbury Festival			Sochi Olympic		
	BL	AD	EX	BL	AD	EX
Raw	171,254	232,811	61,671	213,986	281,692	67,917
Filtered	168,638	215,195	47,274	150,107	205,942	55,980
Proportion	98.32%	92.43%	76.64%	70.15%	73.11%	82.42%

The proportion of event-relevant tweets in both the BL and AD datasets of Glastonbury Festival is higher than that of Sochi Olympic by 28.17% and 19.32% respectively. On the other hand, the figures for EX datasets is reversed. The Glastonbury Festival sees a marked decrease in the retained tweets whereas the Sochi Olympics sees an increase. There are two reasons in response to the lower proportion value for Sochi Olympic in its BL and AD datasets.

- **Tweets with plain text keyword "Sochi"** Although this term represents the city which holds the 2014 winter Olympic games, it also appears in tweets irrelevant to the Winter Olympic since this word is very general. For instance, *Today's Hair: Chanel Brooch, Flora Fresh from Sochi, Weave by @mrericalt, Styling by Mariola!* is collected by both the baseline and adaptive crawlers since it contain the initial keyword "sochi". Due to this reason, when calculating the total score of a tweet (as described in section 3.4.3), the total score of tweet will not change if either "sochi" or its hashtag format "#sochi" emerges in the tweet. However, this results in a significant drop of volume on event-relevant tweets (the volume difference of considering "sochi" as event related versus non-related in BL dataset is 40,038, nearly 20% of the total volume). To reduce the amount of event related tweets that are wrongly removed by the aforementioned strategy while retain the clearness of the filtered dataset, the total score is increased by two when "sochi" and "2014" appears together. This process re-identifies one fourth of the tweets removed by the previous strategy as event-relevant.

- **Tweets in languages other than English** There is no official documentation on Twitter stating that its filtering function retrieves tweets with keyword in other language. However, more than 30 thousands tweets in BL dataset don't contain the initial keywords listed in 4.1 but with the variation of "sochi" and "olympic" in other language. Although some of these tweets escape from the wrongly classification due to event-related hashtags they carried, most of them are filtered out due to the insufficient support on the comparison of Korean characters.

Nevertheless, filtering out the aforementioned event-relevant tweets doesn't bias the investigation of detection benefits in this chapter. First, the event-relevant tweets filtered out by this process are diverse in their content and distribute across the whole investigation period. Second, these tweets are removed from all three filtered datasets. Even if these tweets corresponds to important sub-events, they don't introduce bias to the adaptive crawler since the sub-events are absent across all three datasets.

4.2.3 Parameter tuning

The fundamental idea of this statistical based event detection algorithm is to approximate the subjective observation of the tweets volume. However, social text stream, such as tweets, is always dynamic and hard to anticipate.

There exists four parameters in Twitinfo event detection algorithm which can affect the detection results, as shown in Table 4.3.

TABLE 4.3: Input Parameters of Twitinfo Algorithm

Paramter	Notation	Definition
detection latency	p	number of bins for calculating the initial mean deviation
sample interval	t_{sample}	the length of time slot for each bin in the algorithm
smoothing factor	α	the fraction of recent bin are considered versus all the previous bins
inequality threshold	τ	the threshold to determine whether the variation of tweets volume is big enough to be defined as event

1. **detection latency** The least influential parameter for the algorithm is p. This parameter represents the number of time slots to be used for calculating the initial mean deviation. In other words, it can be considered as the indicator of detection latency for the detection, i.e. a period that is not possible to obtain the detection results immediately. A larger value for this parameter p will result in a longer detection latency but provides a better approximation in the mean deviation. However, the impact of longer detection latency is more substantial then the inaccurate approximation at most of the cases, unless the variation of the tweets

arrival rate at the initial time slots is dramatic. As a result, p is preferable to be a small value, especially when the total detection period is short.

2. **sample interval** When detecting the event through the tweets arrival rate, the length of time slot, or the sample interval (t_{sample}) for calculating the arrival rate, will affect the resolution of the event detection. With a small sample interval, it is possible to detect more events with shorter time span. However, the negative aspect of the shorter sample interval is the increasing of false reports. In order to capture the evolution of the event, an interval that can reveal the key moments during the event is desirable. Namely, the length of sample interval is event specific. For example, the sample interval is about several minutes for a football match while can stretch to hours or even days for disasters like earthquake. The settings of sample interval also impact the p value. A shorter sample interval will results in more significant variations and uncertainty of the tweets count, thus require a larger p.

3. **smoothing factor** The smoothing factor α, an important parameter in the EWMA, is also an essential parameter in Twitinfo algorithm. Similar to the function in EWMA, α determines how many history counts affect the calculation of the current mean average and the mean deviation. The larger the α, the more the mean and deviation is biased towards the recent history. Namely, with the increasing of α, the smooth effect is weakened. There exists a range for α, that is $[0, 1]$. When $\alpha = 0$, the expected arrival rate is just the rescaling of the real arrival rate rescaled by $\frac{C_i - C_1}{p}$. When $\alpha = 1$, the expected arrival rate is only affected by the previous arrival rate. If the event under review requires a small sample interval (less than 5 minutes), a smaller α (less than 0.5) can be helpful in alleviating the impact of false reports. As a result, the instinct for choosing a proper α is to find a value, where the smoothing factor can reduce the impact of trivial variations in tweet arrival rates.

4. **inequality threshold** The threshold τ is another important parameter which is used as a coefficient to determine the variation level of the arrival rate. In the Twitinfo algorithm, a peak window is identified if the difference between the real

rate and the mean rate is significantly higher than τ times of mean deviation. Provided a fixed setting of other parameters, higher τ will results in less peak windows. There is no absolute standard for choosing τ as its value is highly dependent on the value of α. As a result, when choosing the value of τ, it is necessary to find a α, τ pair that can distinguish the visual peaks.

Through the analysis of all the input parameters of Twitinfo's event detection algorithm, it is obvious that the most deterministic parameters are the smoothing factor α and the inequality threshold τ. While the effect of other two parameters can be balanced by changing the value of α and τ.

Although it is possible to adopt the Twitinfo system's setting directly ($p = 5$, one-minute sample interval, $\alpha = 0.125$, $\tau = 2$), a universal parameter setting for all the events is not achievable. This is because the variation of tweets volume is event specific [98]. As a result, this research determines Twitinfo's parameters in a heuristic strategy by considering the characteristics of each event with statistical and empirical observations.

This research chooses p to a value that cover a constant period of time, i.e. 30 minutes, for all the evaluation. By employing a fixed time period, the p value is negative variant to the length of sample interval t_{sample}. An empirical experiment on different sample interval proves that the intensive variation of volume brought in by shorter time interval can be balanced by providing the algorithm with a small α. In other words, when the sample interval is decreased, α need to be reduced for obtaining a similar visually result. Consequently, the sample interval is determined on event basis: 5 minutes for Glastonbury Festival as this is the average length of a song during the performance, 10 minutes for Sochi Olympic as the shortest final program last for that period of time. For the setting of α and τ, this research explores the number of window that can be detected for each pair of them. A ground truth for the location and length of peak is generated based on visually perception on the volume. By comparing the detected windows against the ground truth, the parameters are set with values that enable the highest detection accuracy (all the bursts are detected even after smoothing).

In this chapter, the parameter is tuned for the BL dataset. The same values are applied

to the other two datasets, i.e. AD and EX. Namely, the parameters are determined with tweets arrival rate of the BL dataset, and therefore may not be the best choice for the AD or EX datasets. However, if the detection result over AD and EX datasets are improved even using the compromised parameters (particular tuned for BL), it is rational to deduct that the result should be the same or even better if the parameters are tuned with those datasets. Table 4.4 lists the configuration of parameters for the event under investigation.

TABLE 4.4: Parameters Setting for Twitinfo Algorithm

	Parameters			
	p	t_{sample}	α	τ
Glastonbury Festival	6	5mins	0.6	2.5
Sochi Olympic	3	10mins	0.75	2.75

4.2.4 Evaluation Metrics

The majority of the event detection algorithms are evaluated by their ability to successfully identify real-world events (i.e. precision and recall as described in section 2.4.1). However, rather than comparing the detection results generated by different detection algorithms, this thesis is interested in exploring the detection results generated by the same detection algorithm but over different event datasets. To be more specific, this chapter aims at measuring the differences between the detection results that are generated by the Twitinfo algorithm but over BL, AD and EX datasets. Accordingly, the hypotheses for the research questions that proposed at the beginning of this chapter can be concluded as:

- **Hypothesis 1**: The accuracy of detection results is improved when the additional event relevant content is introduced.
- **Hypothesis 2**: More sub-events can be detected from the datasets that contain extra event content
- **Hypothesis 3**: The amount of information carried by the sub-events which are detected from datasets with extra event content is higher than that from BL datasets

This research quantifies the detection accuracy by the precision, recall and F-measure (in section 4.2.4.1), correlated to the amount of reasonable sub-events that can be detected from the datasets. One the other hand, the amount of information in the detected sub-event is quantified by the event entropy (in section 4.2.4.2).

4.2.4.1 Detection Precision, Recall and F-Measure

The raw output of the Twitinfo algorithm is a list of peak windows. Assessing the performance of Twitinfo event detection algorithm solely with the detected peak windows can be very difficult. Even though the peak windows from different datasets overlap with each other, the contents covered by these windows can be different since the tweets in different peak windows varies.

To simplify the comparison of peak windows between BL, AD and EX datasets, this research relies on the summary tweets (as introduced in section 4.1.3) for measuring the detection accuracy. The measurements for investigating the detection results are shown in Table 4.5. Rather than checking the number of realistic event can be detected from the noisy Twitter stream, this research examines the detection accuracy over a collection of tweets that are associated with a particular event. Therefore, applying the event detection algorithm to the retrieved datasets should provide peak windows about the sub-events (the number of sub-events is normally lower than the number of peak windows can be detected from an event stream). These sub-events reveal the evolution of the event.

This research uses **event precision** (P_{event}) to measure the proportion of peak windows that correspond to the real-world sub-events. As described in the section 2.1.1, a sub-event of an event refers to the underlying story that happens at a particular time period. In the scenario of this research, a sub-event is newsworthy to be reported by the mainstream media, showing together with the retrieved keywords in the headline or content. For example, a peak window that is summarised by a tweet "*I think I'd be quite into Glastonbury if I was some kind of predator or serial killer*" in the Glastonbury Festival datasets will not be considered as a sub-event. Clearly, this is an opinion

tweet and doesn't corresponding to any realistic events. This research defines the **event recall** (R_{event}) as the proportion of distinct sub-events can be correctly identified by the framework. When applying the Twitinfo algorithm on the event datasets, there is chance to detect multiple peak windows (normally consecutively) talking about the same sub-event. If two detected peak windows are related to the same sub-event, both of them will add credit to the P_{event}, but only one distinct sub-event will be considered when calculating event recall. Since it is infeasible to label the nearly half million tweets manually for identifying all the sub-events, the number of distinct sub-events is defined as the total number of distinct sub-events can be detected from all three (BL, AD and EX) datasets that are about the same event.

TABLE 4.5: Detection Precision and Duplicate Rate

Notation	Definition
P_{event}	fraction of peak windows that are realistic sub-event
R_{event}	fraction of distinct sub-events that are detected as peak windows
F_1 score	considering the above P_{event} and R_{event} by equation 3.7

4.2.4.2 Event Entropy

In order to verify the third hypothesis, it is necessary to quantify the amount of information that can be detected from each event. This research argues that the summary tweet of a window elaborates the most important sub-event but cannot represent the overall state of the window. On the contrary, the summary terms are more representative of the content of their window since they are carried by multiple different tweets within the peak window. The summary terms of each peak window are exploited for measuring the amount of information that can be detected from an event. Shannon entropy, also referred to information entropy [128], is used since it has been used in existing event detection research. A Higher entropy indicates larger amounts of information in an event cluster [32, 33]. The information entropy measures the uncertainty of information content based on the assumption that a set with random symbol can provide more information. The information entropy of a message can be calculated by equation 4.6,

where $P(x_i)$ represents the probability that symbol x_i occurs.

$$H(X) = \sum_i P(x_i)I(x_i) = -\sum_i P(x_i)\log P(x_i) \tag{4.6}$$

According to the equation in 4.6, information entropy is affected by two factors: the probability distribution of all the symbols and the number of symbols. For a message of certain number of symbols, the entropy reaches its maximum when all the symbols equally emerge. Increasing the number of terms will also result in higher entropy. For this research, a summary term is judged as more informative when it is associated with diverse non-stop-word terms.

This research measures the event entropy by calculating the entropy of each summary term. Based on the entropy formula in equation 4.6, it is possible to quantify the amount of information that a summary term represents. For each summary term $t \in T$ of peak window $w \in W$, it represents all the tweets which contain this term t and locate within window w. This set of tweets is represented by TS. Consequently, the summary term entropy for term t is equivalent to the entropy TS. Namely, the amount of information that is carried by summary term t of window w can be calculated by equation 4.7, where x_i is the distinct term in tweet set TS.

$$H(t,w) = H(TS) = -\sum_i P(x_i)\log P(x_i) \tag{4.7}$$

As a result, the amount of information of window w, or the window entropy, is calculated by the sum of all the summary term entropy, as shown in equation 4.8

$$H(w) = \sum_{t \in T} H(t,w) \tag{4.8}$$

The sum of window entropy of all the detected windows in W can measure the information entropy of an event e. However, the number of peak windows among BL, AD and EX datasets for the same event can be different. This research calculates the average window entropy instead of accumulative window entropy to measure the event entropy.

In summary, this research quantifies the information that can be detected for event e,

or the detectable event entropy, by equation 4.9.

$$H(e) = \frac{1}{n} \cdot \sum_{w \in W} H(w) = \frac{1}{n} \cdot \sum_{w \in W} \sum_{t \in T} H(t, w) \tag{4.9}$$

where n is the number of window within the detected windows set W.

4.3 Investigating DSTReaM with Adaptive Datasets

This section reports the event detection results based on the evaluation metrics listed above. Two separate experiments are conducted to investigate the event detection results on BL, AD and EX datasets over Glastonbury Festival and Sochi Olympic events. In the first experiment, the detection algorithm is applied to the unfiltered raw datasets, while in the second experiment, the datasets are filtered to retain only event-relevant traffic. For both events, tweets are provided to the detection algorithm in a continuously streaming manner to simulate the real-time event detection scenario. To conclude the investigation results, a discussion on the performance of evaluation metrics and the characteristics of detection algorithm is given at the end of this section.

4.3.1 Experiment One: Detection Results over Raw Datasets

To mine sub-event information about the event in real-time, the event detection algorithm should be able to analyse the tweets data in streaming manner without additional pre-processing. As a result, this research first applies the detection algorithm to the unfiltered raw data from the Twitter crawlers. Namely, the entire dataset, including both event-relevant traffic and event-irrelevant tweets are counted and analysed for generating the peak window and summary.

4.3.1.1 Peak Windows and Sub-events

The event streams (reproduced from BL, AD and EX datasets) are sampled and sent to the Twitinfo detection algorithm, whose parameters are configured to the value in Table

4.4. Based on the criteria in section 4.2.4, sub-events are identified from all the detected peak windows (an example of detection result can be found in Table B). Specifically, each window and its summary tweet is examined by at least two participants. Their task is to compare the summary tweets with the online resources from event websites, mainstream media and Wikipedia pages. If there exists a report or Wikipedia page indicating the association between the entities mentioned in the summary tweet and the event, the peak window described by the tweet is considered as a sub-event.

In this section, the full list of sub-events and the detected window are presented by a summary table with visualisation. The full list of sub-events that are detected from the Glastonbury Festival and Sochi Olympic are reported in Table 4.6 (Raw columns) and Table 4.9 (Raw columns) respectively. While the description for noisy peak windows (which not correspond to sub-event) emerged in Glastonbury Festival and Sochi Olympic are reported in Table 4.7 (Raw columns) and Table 4.8 (Raw columns) respectively. A check mark is given when the sub-event or noisy peak window is detectable with that particular dataset. The Keyword column indicating the hashtags that are automatically identified as tracking keywords during the retrieval process. Similarly, the visualisations of peak windows for these two events are presented separately. Figure 4.2 visualises the peak windows detected from Raw Glastonbury datasets, while Figure 4.3 visualises the peak windows detected from Sochi Olympic datasets. The figure of each event visualises the detected peak windows by multiple boxes. Lettered boxes represents the noisy peak windows. The rest boxes (indexed box) are labelled by the index of sub-event with which their summary tweet is associated.

TABLE 4.6: Description of Sub-Events (Glastonbury Festival)

Sub-event Idx	Event Title	Raw			Filtered			Keyword
		BL	AD	EX	BL	AD	EX	
1	Billy Bragg performance						✓	
2	Ben Howard performance			✓	✓	✓	✓	#benhoward, #onlylove
3	Laura Mvula performance	✓		✓	✓	✓		#lauramvula
4	Tibetan Monk Throat Singing	✓						
5	Elvis Costello performance	✓						#elvis, #elviscostello
6	Interview a Hula Hoop therapist				✓	✓		
7	Noah and the Whale performance		✓	✓	✓	✓	✓	#noahandthewhale
8	Alabama Shakes performance						✓	#alabamashakes
9	Primal Scream performance			✓			✓	#primalscream
10	Maverick Sabre performance			✓		✓	✓	#mavericksabre
11	Glastonbury founder supports badger cull	✓	✓		✓	✓		#badgers, #badgercull, #badgerswagger, #stopthecull
12	Two Door Cinema Club performance		✓	✓			✓	#TDCC, #twodoorcinemaclub
13	Example performance			✓				#TDCC, #twodoorcinemaclub
14	Rollingstones performance	✓	✓	✓	✓	✓	✓	#stones, #therollingstone, #rollingstones, #thestones, #stonesglasto

TABLE 4.7: Description for Noisy Peak Windows (Glastonbury Festival)

Noise Idx	Event Title	Raw			Filtered		
		BL	AD	EX	BL	AD	EX
A	tweets about Glastonbury Festival but express personal opinion or are not reported in news	✓	✓	✓	✓	✓	✓
B	Wimbledon Championship		✓	✓			
C	Twitter follower and followee advertisements		✓	✓			
D	Discussion about Manchester United F.C.[†] and football games			✓			

[†]a professional football clubs that based in Manchester, UK

TABLE 4.8: Description for Noisy Peak Windows (Sochi Olympic)

Noise Idx	Event Title	Raw			Filtered		
		BL	AD	EX	BL	AD	EX
A	Iraq women protection propaganda	✓	✓	✓			
B	A general discussion about Olympic Women's hockey	✓	✓		✓	✓	

Table 4.9: Description for Sub-Events (Sochi Olympic)

Sub-event Idx	Event Title	Raw			Filtered			Keyword
		BL	AD	EX	BL	AD	EX	
1	Tomoka Takeuchi qualification for snowboard parallel				✓	✓		#snowboard,
2	Kim Yuna's Retirement Announcement	✓	✓					#yunakim, #yuna,#majesty
3	Figure skting gala show					✓	✓	#gala, #yunakim, #yuna,#majesty, #figureskating
4	Ice hockey Canada vs. USA	✓	✓		✓	✓		#canadahockey, #usahockey, #usavscanada, #canvsusa, #icehockey
5	Julia Dujmovitis weman's snowboard champion				✓		✓	#dujmovits, #snowboard, #snowboarding
6	Vic Wild for men's snowboard champion			✓			✓	#vicwild, #snowboard, #snowboarding
7	Photo of the day by three Olympic champions			✓		✓	✓	
8	Asahi Shimbun: Mao Asada feature report	✓	✓					#maoasada, #figureskating
9	Alexis Pinturault during Alpine skiing competition					✓		#pinturault, #ski, #slalom
10	Speed skating champion by Netherland			✓		✓		#speedskating
11	Plushenko Back Surgery		✓					
12	Anton Shinpulin played in Russain biathlon team, won the relay champion	✓	✓	✓	✓	✓	✓	#biathlon, #fourcade
13	Ice hockey USA vs. Finland	✓	✓	✓	✓	✓	✓	#finvsusa, #hockeybest

As can be observed from the *Raw* columns from Table 4.6, the number of distinct sub-events can be detected from BL datasets is higher than that in AD datasets, while both lower than the number of sub-events detected from the EX datasets. However, both the AD datasets and the EX datasets provides sub-events that can't be detected from the BL datasets. On the other hand, both AD and EX datasets cover more noisy peak windows in their detection results (as shown in Table 4.7). These noisy peak windows are not limited to opinion chat about Glastonbury Festival but also talk about events such as Wimbledon Championship, Twitter follow-up scam and discussion about Football club. However, as can be observed from Figure 4.2, these sub-events are detected during the period when the amount of traffic is high during Glastonbury Festival. This further demonstrates the conclusion from section 3.6.2.4 that the KwAAs give better performance during performance period. In addition to the extra sub-events, CS-KwAA also identifies keywords relating to sub-events. More than 80% of the sub-events detected in the Glastonbury Festival can be described with the identified keywords (as shown in *Keyword* column from Table 4.6).

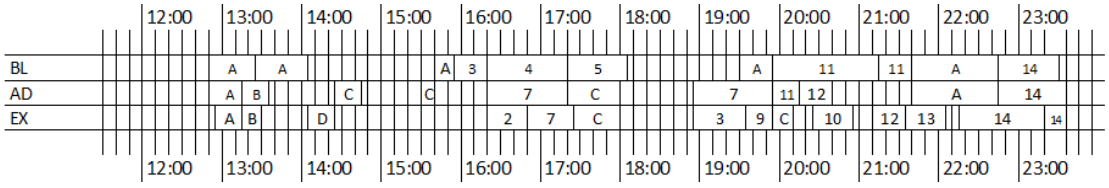


FIGURE 4.2: Detected Peak Windows of Glastonbury Festival (Raw):
Numbered windows are indexed by Sub-event Idx. defined in Table 4.6;
Lettered windows are indexed by Noise Idx. defined in Table 4.7

As show in Figure 4.2, the amount of noisy peak windows identified during the Sochi Olympics (lettered boxes) is decreased. In fact, there only exists two types of noisy peak window for the Sochi Olympic datasets, as shown in Table 4.8. Both of them are detected in the BL datasets since the initial keyword "Sochi2014" is mentioned. The improves on the detection results of sub-events also provides much better statistic than that in Glastonbury Festival test. As shown in Table 4.9, the AD dataset provides not only all the sub-events that are detected from BL datasets, but also an additional sub-event about *Plushenko*. The EX datasets provides less sub-events that emerged in the peak windows detected from BL datasets, but offers three additional sub-events that

neither detectable from BL datasets nor the AD datasets. These can be considered as the supplementary materials for revealing the evolution of Sochi Olympic as more time slots are filled.

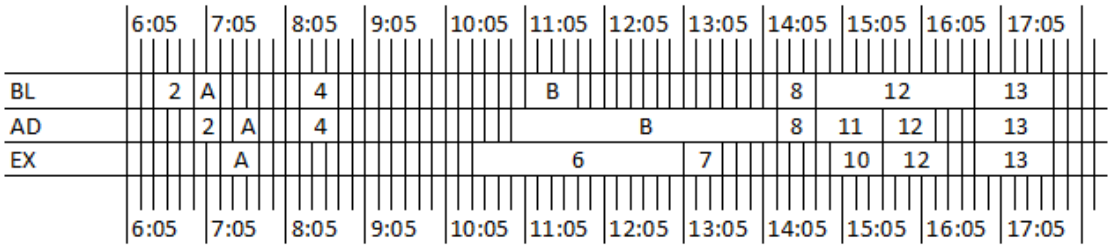


FIGURE 4.3: Detected Peak Windows of Sochi Olympic (Raw): Numbered windows are indexed by Sub-event Idx. defined in Table 4.9; Lettered windows are indexed by Noise Idx. defined in Table 4.8

4.3.1.2 Detection Accuracy and Event Entropy

In order to quantify the metrics about detection accuracy and the event entropy, the sub-events (numbered peak window in Figure 4.2 and 4.3) identified from whole list of the peak windows are examined. Table 4.10 reports the number of peak windows that can be detected from different type of datasets (which belongs to different events). The BL and AD datasets of Glastonbury Festival results in the same amount of peak windows, while both are less than that of EX dataset (11 versus 14). 6 out of 11 peak windows in BL datasets correspond to the realistic sub-events (as illustrated in Figure 4.2). This results in the 49.59% F_1 score in BL dataset. The F_1 score for the AD dataset is lower than that of BL by 9.19%. In fact, it is the lowest among all three Glastonbury datasets. This is caused by event-irrelevant tweets. They decrease both the detection precision and recall. Among all the 11 peak windows of the AD dataset, 5 of them are labelled as realistic sub-event, 2 of them talks about Glastonbury but can't be associated to the reports from online sources. The rest 4 are noisy windows that are irrelevant to the Glastonbury Festival (talking about Wimbledon Championships, Premier League and etc.). Although the amount of noise brought in by the adaptive crawling is less than 1% of the event-relevant traffic (as shown in Table 3.8), the Twitinfo event detection algorithm still identifies these as abnormal moments (outliers). As described in section

3.6.2, CS-KwAA can quickly recover from non-related event keywords. However, this advantage negatively impact the detection accuracy on the sub-event detection task. Twitinfo algorithm is very sensitive to the sudden spikes caused by the short-lived event non-related keywords. Due to that, the AD dataset gets the lowest event recall. EX dataset gets the highest F_1 score. Although the number of sub-events detected in EX dataset is larger than that of BL by one, the detection precision is lower due to non-Glastonbury peak windows. However, all eight sub-events are distinct and result in the highest event recall.

TABLE 4.10: Evaluation Metrics on Raw Datasets

	Glastonbury Festival			Sochi Olympic		
	BL	AD	EX	BL	AD	EX
Windows No.	11	11	14	7	8	6
P_{event}	54.55%	45.45%	64.29%	71.43%	75%	83.33%
R_{event}	45.45%	36.36%	72.73%	55.56%	66.67%	55.56%
F_1 score	49.59%	40.40%	68.25%	62.50%	70.59%	66.37%
Event Entropy	23.38	30.09	32.02	21.95	24.61	31.19

When quantifying the event entropy, this research only considers peak windows that are identified as sub-events. As shown in Table 4.10, although the F_1 score calculated based on detection results of AD dataset is the lowest , the amount of information carried by those limited number of sub-events are substantially higher (by nearly 30% increasing) than the event windows of BL datasets. The event entropy of EX dataset is higher than both BL's and AD's, indicating that the amount of information carried by the extra information is the main contributor to the increasing of event entropy for AD dataset.

The event entropy metric shows similar tendency for the Sochi Olympic datasets, highest for the EX dataset while lowest for the BL dataset. The F_1 score for Sochi Olympic shows different scene, but all are higher than that of Glastonbury Festival. The reasons for this change are twofold. First, unlike the generic expression about the willingness to go to the festival in the Glastonbury datasets, tweets containing personal feeling in Olympic datasets are always associated with a real-world entity, such as a team or an athlete who is playing the game. Since these tweets are about things reported in newswire, the probability that a peak window is a sub-event is larger. Second, the sub-

events in Olympic datasets happen chronologically rather than simultaneously and thus are easier to identify. Comparing the F_1 score across the detection results based on all three datasets, it reaches the highest based on AD dataset, the second place is based on EX dataset, while the last is based on BL dataset. Contrary to the Glastonbury Festival, the benefits of additional event traffic is more obvious by the results of AD dataset rather than EX dataset. The enhancement of the results in AD dataset is owing to the high proportion of event-relevant tweets (as shown in Table 4.2). As a result, no additional noisy windows are detected compared to the results of BL dataset. The main reason for the lower F_1 score in EX dataset is the number of peak windows, this number is even lower than that in BL dataset.

4.3.2 Experiment Two: Detection Results over Filtered Datasets

As shown in the previous experiment, the detection results on the AD and EX datasets are affected by the event-irrelevant tweets, especially for the Glastonbury Festival datasets. Therefore, the second experiment investigates the detection results on datasets that contain only event-relevant tweets. As described in section 4.2.2, the keywords are labelled to guide the tweets classification. This research then follows the same steps as the experiment one, but investigating the detection results based on the *filtered* datasets of Glastonbury Festival and Sochi Olympic.

4.3.2.1 Peak Windows and Sub-events

Following the same steps in section 4.3.1.1, the description for sub-events and noisy peak windows for the filtered datasets are generated and visualised. Figure 4.4 illustrate the detection results of all three filtered datasets about Glastonbury Festival (the descriptions for all the boxes in the figure are listed in Table 4.6 *Filtered* columns and Table 4.7 *Filtered* columns), while Figure 4.5 presents the results detected from the filtered Sochi Olympic datasets (the description for each box in the figure can be found in Table 4.9 *Filtered* columns, Table 4.8 *Filtered* columns).

As shown in the Filtered columns in Table 4.6, one additional sub-event is detected from

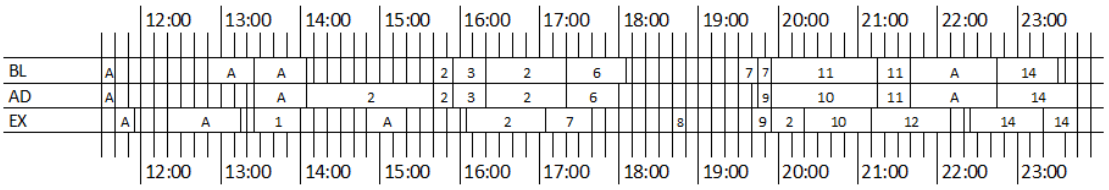


FIGURE 4.4: Detected Peak Windows of Glastonbury Festival (Filtered): Numbered windows are indexed by Sub-event Idx. defined in Table 4.6; Lettered windows are indexed by Noise Idx. defined in Table 4.7

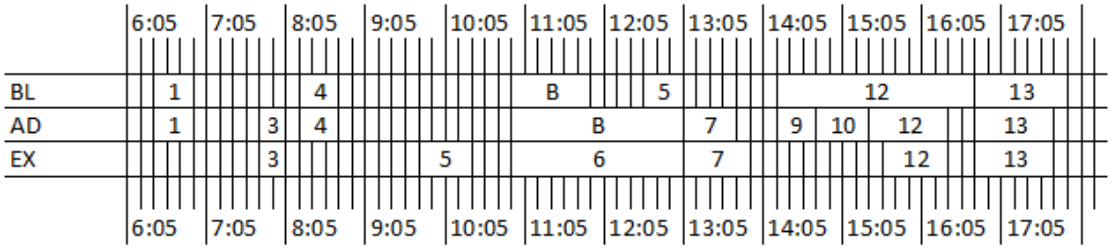


FIGURE 4.5: Detected Peak Windows of Sochi Olympic (Filtered): Numbered windows are indexed by Sub-event Idx. defined in Table 4.9; Lettered windows are indexed by Noise Idx. defined in Table 4.8

the AD dataset even with the parameter tuned for the BL datasets. When applying the event detection algorithm only on the extra traffic, much more underlying events can be detected. 5 more sub-events are detected comparing to the results of BL. Although the AD datasets contains all the traffic in EX datasets, sub-events 1, 8, 9 and 12 are absent from the detected result of AD dataset. In fact, overwhelming by the huge volume of tweets about other more trending sub-events in AD datasets, these four absent events are missed by the detection algorithm. As shown in Figure 4.4, the windows of two distinct sub-events in EX (talking about the performance of Primal Scream and Two Door Cinema Club) overlap with peak windows in AD dataset. Similar to the results based on raw datasets, most of the sub-events detected from filtered datasets can be described by the tracking keywords. As reported in Table 4.6 (Filtered and Keyword columns), nine of the eleven sub-events are tracked with keywords identified by the proposed KwAA. Namely, it is possible to get these sub-events in real-time with these keywords while crawling.

A more substantial difference of the detection results among BL, AD and EX datasets can be found in the investigation of Sochi Olympic. The *Filtered* column in Table 4.9

and Figure 4.5 illustrates the number of additional sub-events can be detected from AD or EX datasets when compared to that from the BL dataset. Using BL dataset as reference, AD dataset provides 4 more sub-events and EX dataset provides three. Rather than reveals more underlying, overlapped sub-events (which is demonstrated by the analysis on event detection over Glastonbury Festival datasets), the AD dataset of Sochi Olympic provides better resolution on event detection. During the period where the Twitinfo algorithm detects the Biathlon relay competition (sub-event 9) in BL dataset, the filtered AD dataset presents 3 separate sub-events within that period. Similarly, sub-events detected by CS-KwAAs during Sochi Olympic can be represented with keywords identified by CS-KwAAs.

Based on the above analysis, it can be observed that the detection results based on the Filtered datasets is different from the results based on the Raw datasets. Even though the same parameters settings in Table 4.4 are adopted, the algorithm detects new sub-events that are not recognized in the previous experiment. On the other hand, the number of noisy peak windows is reduced after filtering out the event-irrelevant tweets. This is more apparent for the Glastonbury Festival. Three types of noisy peak windows are absent in the filtered datasets. The only remaining one is about the event.

4.3.2.2 Detection Accuracy and Event Entropy

To compare the detection results among raw and filtered datasets, this experiment investigates the results based on the same metrics in Table 4.10. Based on the detected peak windows and the list of detectable sub-events for Glastonbury Festival datasets (as shown in Table 4.6 and Figure 4.4) and Sochi Olympic datasets (as shown in Table 4.9 and Figure 4.5), the detection precision, recall, F measure and event entropy are calculated and listed in Table 4.11.

As can be seen from Table 4.11, the F_1 score for AD and EX datasets are higher than that of BL for both events, by 7.83% and 9.92% respectively for Glastonbury Festival and by 22.21% and 12.50% respectively for Sochi Olympic. For the filtered datasets, all the detected peak windows are about the event of interest, but only some of them

TABLE 4.11: Evaluation Metrics on Filtered Datasets

	Glastonbury Festival			Sochi Olympic		
	BL	AD	EX	BL	AD	EX
Windows No.	13	12	13	6	10	6
P_{event}	69.23%	75.00%	76.92%	83.33%	90.00%	100.00%
R_{event}	54.55%	63.64%	72.73%	50.00%	80.00%	60.00%
F_1 score	61.02%	68.85%	74.77%	62.50%	84.71%	75.00%
Event Entropy	25.93	28.02	26.87	25.46	26.20	28.07

correspond to the realistic sub-events. All the datasets of Glastonbury Festival results in 9 sub-event windows, while the number of peak windows corresponding to sub-event varies for Sochi Olympic. There are 5, 9 and 6 sub-event windows for BL, AD and EX dataset respectively. In other words, the amount of sub-events from the AD and EX datasets are at least equal to that in the BL datasets for both events. Consequently, both the detection precision and recall are higher for the datasets which contain extra event traffic, indicating that the adaptive crawling can bring benefit to the event detection task.

The other metric, event entropy follows the same pattern as that for the raw datasets. The amount of information in AD and EX datasets is still higher than that of the BL datasets for both of the events. However, the differences between these become less obvious due to the removal of noisy tweets.

4.3.3 Discussion

By comparing the results in Table 4.10 and Table 4.11, it is clear that the detection accuracy is improved on both the AD and EX datasets for both Glastonbury Festival and Sochi Olympic event. In a nutshell, the improvement on F_1 score is achieved by the higher proportion of sub-events across all the peak windows and lower proportion of duplicated sub-events. The proportion of sub-events (the detection precision) for EX dataset of Sochi Olympic even reaches 100%, indicating that all the detected windows corresponds to realistic sub-events. However, with higher proportion of event-relevant tweets, sub-events tend to be detected in duplicate. The number of duplicate sub-events increases from 1 to 6 for Glastonbury Festival and from 0 to 2 for Sochi Olympic. On the

other hand, filtering out irrelevant tweets doesn't impact the F_1 score on BL datasets. Since all the baseline tweets contain the initial keyword, the proportion of noisy tweets in BL datasets is small (normally less than 5%). As a result, the F_1 score of BL dataset maintains the same level for both events. While the improvement of the F_1 score is 7% for EX datasets and 14% for AD datasets, the maximum increasing for BL datasets is 3%. However, the overall situation is the improvement on the F_1 score, precision and recall after filtering. This indicates that the filtering process bring positive impact on the event detection tasks, especially for datasets retrieved by adaptive crawling.

The analysis on the decomposed stream illustrates that the Twitinfo peak detection algorithm can identify extra sub-event with the help of the extra event tweets. This is validated by both the raw datasets and the filtered datasets of two different events. To determine whether the DSTReaM framework performs better in sub-event detection, the detection results on Glastonbury Festival and Sochi Olympic with both raw and filtered dataset are examined (as shown in Table 4.12 and Table 4.13).

Table 4.12: DSTReaM on Raw Datasets

	Glastonbury Festival		Sochi Olympic	
	$max(\cdot)$	ALL	$max(\cdot)$	ALL
Windows No.	14	28	8	15
P_{event}	64.29%	58.62%	83.33%	80.00%
R_{event}	72.73%	90.91%	66.67%	100.00%
F_1 score	68.25%	71.28%	70.59%	88.89%

Table 4.13: DSTReaM on Filtered Datasets

	Glastonbury Festival		Sochi Olympic	
	$max(\cdot)$	ALL	$max(\cdot)$	ALL
Windows No.	13	26	10	13
P_{event}	76.92%	73.08%	100.00%	76.92%
R_{event}	72.73%	81.82%	80.00%	100.00%
F_1 score	74.77%	77.20%	84.71%	86.95%

In the tables, the $max(\cdot)$ column represents the max value of the metric among BL, AD and EX in the corresponding table. For example, the event precision of $max(\cdot)$ for Glastonbury Festival in Table 4.12 is the maximal event precision for the same event in Table 4.10. As shown in the two tables, by using the proposed DSTReaM framework (i.e.

ALL column), the recall and F1 score are improved for all cases. Although the precision drops due to the larger number of peak windows after combination, the detection recall for all the experiments using DSTReaM framework is higher. As a result, the F1 score is also improved for both events. The DSTReaM framework outperformed the state-of-the art event detection algorithm in providing more sub-events. It can be concluded that the proposed parallel detection framework introduces improvement compared with using any one data stream (i.e. BL, AD or EX) alone.

A further result of the experiments in this chapter is that statistical outlier detection based event detection algorithms are very sensitive to noise. This issue is even more severe on the datasets crawled by the adaptive crawler with CS-KwAA. As discussed in section 3.6.1, the advantage of CS-KwAA is its quick recovery from wrongly identified event non-related keywords. However, this actually becomes the major cause as this mechanism introduces apparent noisy outliers: every time when a noisy keyword is quickly dropped, a burst is likely to be generated and detected. Besides, the algorithm employed in this chapter is fond of events with chronological sub-events. When applying it to events containing simultaneous sub-events, the Twitinfo algorithm lost the underlying, less trending sub-events. Although the Twitinfo algorithm raises additional challenge to this investigation, the proposed adaptive crawling not only provides topical keywords on the core of sub-events, but also provides datasets with additional sub-events that can't be detected from baseline datasets.

4.4 Summary

By proposing the DSTReaM, this chapter investigates a different perspective that is left to be unexplored in the existing literatures: the effects and influences of the extra event traffic on the event detection algorithm. For the proposed framework, the input is the expanded event stream that is identified by this research's novel CS-KwAA embedded adaptive crawler and decomposed into separate streams to be analysed in parallel by the Twitinfo peak detection algorithm, before being recombined to identify and summarise sub-events. In order to understand the impact of additional event information, this chap-

ter investigated the performance of DSTReaM over two different planned events using the metrics of detection accuracy and detection entropy. This chapter demonstrated in two events of a distinct and diverse nature that the DSTReaM provides better event detection in three primary aspects:

- a higher recall and F1 score. This demonstrates that the adaptive crawler introduces additional sub-events that are not detectable by other TEM systems.
- a higher event entropy for the adaptive datasets than the baseline datasets. With larger amount of event information carried by adaptive datasets, the tweets describe the event and sub-events are diversify in the vocabulary
- keywords describing the sub-events that are detected by Twitinfo algorithm. These descriptive keywords are identified by CS-KwAA during the collection of events tweets.

On the other hand, the investigation of the Twitinfo algorithm also shows the deficiencies of algorithm when processing the adaptive datasets. It is possible to monitoring the event with better sub-event detection if the algorithm is capable to:

- overcome the false detection by automatically identification of the noise in real-time
- adapt the detection resolution based on the characteristics of events and underlying sub-events
- monitoring each sub-event separately in a hierarchy mode so to detect sub-events occurring simultaneously or overlapping in time

In addition, as CS-KwAA identified keywords that are related to sub-events during the tweets retrieval, it is possible to get the sub-events, without applying the detection algorithm, solely from the keywords identified by the proposed CS-KwAA.

Chapter 5

Conclusion and Future Work

In a world where the majority of citizens have mobile phones with embedded cameras, microphones and sensors that can access to the internet, the rate at which data can be produced during an event and disseminated is therefore increasing dramatically. This phenomenon, together with the emergence of online social media, is changing the way that people engage with events. Rather than consume event news passively as a reader, in this Web 2.0 era, each individual has a chance to act as a the journalist for some headline news too. With richer and more immediate information about real-word events available from Twitter and other microblogging services, new opportunities and challenges are arising to enhance the use of the conventional TDT (Topic Detection and Tracking) solutions. Over the last decade, a notable research effort has been made to apply TDT solution to online social media. Twitter, as the most newsworthy platform among the popular online social media services [27], receives the most attention. To improve event monitoring through Twitter, researchers tend to extend the depth of detection by developing algorithms that are capable of detecting as many realistic events as possible (as described in chapter 2). The work described in this thesis concerns the similar problem but for an online microblog setting. Rather than relying on sophisticated but inefficient algorithms that improve the Twitter event detection problem through the depth of detection, this thesis explores the feasibility of improving event monitoring by expanding the coverage of event-relevant tweets. This chapter first summarises the work

in this thesis (in section 5.1) and concludes with recommendations to extend this work (in section 5.2).

5.1 Conclusion

To enable a better coverage of event-relevant content from microblogs, this thesis first proposes an adaptive crawling model in chapter 3. In order to allow the model to run in a fully automatic manner without requiring human annotation, this thesis proposed to identify extra event content by relying on tweets that were retrieved previously. Specifically, the proposed adaptive crawling model enables additional event-relevant content to be retrieved by identifying additional topical keywords as the search terms. By exploiting only the Twitter # symbol that emerged in previous tweets, the adaptive crawling model enables the whole process to automatically run in real-time and copes with both planned and unplanned events.

In order to improve both the retrieval precision and recall, three KwAAs are proposed for expanding the search query and the coverage of event content (in chapter 3). By applying the adaptive crawlers equipped with different KwAAs in multiple real-world events, this thesis evaluates the proposed model and KwAAs against the datasets retrieved by a baseline crawler. This baseline crawler retrieves event tweets based on a set of predefined keywords. Though the baseline crawler collects data in a straightforward way, it is employed in most of the existing Twitter event analysis research. To avoid the bias towards a good performance when using a priori knowledge of specific events, this research selects the most general event term, normally in plain text, as the initial search keywords. The experiment on the Glastonbury Festival datasets shows that the high frequency hashtags introduce both event non-related keywords and event irrelevant tweets. In contrast, the other two KwAAs achieve more promising results since the precision and recall of the identifying underlying event topics are improved. Compared with the algorithm based upon Twitter's traffic pattern, hashtags with high content similarity to the initial keywords are more reliable for retrieving the event topics. Based on the evaluation of all the KwAAs with a real-world event, this research concludes that

the adaptive crawlers equipped with traffic pattern and content similarity KwAAs can identify a notable amount of event-related keywords, and thus contribute to a greater amount of event-relevant tweets.

With a better coverage of event keywords and event tweets, the proposed adaptive crawling mechanism and KwAAs are adopted and extended for building a domain specific (crisis) lexicon [88]. While the proposed model and algorithm are capable of dealing with certain events, the problem of concern is the generality of using them with different kinds of events. As a result, a further evaluation is carried out on the KwAAs across four different real-world events: two planned events, 2013 Glastonbury Music Festival and 2013 Wimbledon Tennis Championship, and two unplanned events, 2011 Egypt Protests and 2014 (MH370) Missing Plane. These events have different characteristics. The results show that the proposed KwAAs have a better performance for planned events in terms of topical keyword identification and information to noise ratio on the extra traffic. In addition, the performance is enhanced during the actual event period of planned events. As for the unplanned events, the performance of both traffic pattern and content similarity based KwAAs degrades. However, the algorithm based on content similarity still gives a much better performance than that based on traffic patterns. This indicates that the content of tweets is a more reliable feature for discovering extra event relevant content.

This thesis also presents a TEM solution, i.e., DSTReaM, that helps to verify that the extended coverage of event content contributes to improved event monitoring (in chapter 4). To investigate the effects and influences of extra event traffic on the event detection algorithm, the DSTReaM first identifies event content by using the Adaptive Microblog crawler (as described in chapter 3). Then, the input stream is decomposed and analysed. Specifically, the framework decomposes the input stream into three individual sub-streams, and parallels a statistical-based peak detection algorithm over the temporal features of those sub-streams. Based on the information that is extracted, newsworthy sub-events are detected and summarised. The framework is validated with two different planned events using metrics based upon detection accuracy and event entropy. Although the Twitinfo detection algorithm is sensitive to its parameter settings

and noise, the detection accuracy and detectable event entropy metrics are improved. In other words, it is shown that better event detection can be achieved if the coverage of the event content is expanded. Besides, this chapter, 4, also demonstrates that the potential of using KwAA for sub-event detection in real-time. By comparing the summarised event with the keywords that are identified by the CS-KwAA, this research reveals that the search terms cover most of the detected sub-events (in section 5.2).

5.2 Future Work

To enable better event awareness and to monitor and report events in a real-time manner, some recommendations of future work based on the research in this thesis are given as follows:

1. **Refinements on the adaptive crawling model with A) text modelling based on ontologies, B) automatic pre-defined keywords identification and C) event stream noise reduction (extension of chapter 3)**

    - In this thesis, the proposed crawling model expands queries purely based on the text content of tweets. However, text content in Twittersphere is full of typos and has lexical variations. To reduce the impact of semantic issues, existing research employs Semantic web technologies such as ontologies, for better suggestions based upon query expansion [129]. For example, ontology provides a formal structure, including the types, properties, and interrelationships, of the entities that exist for a particular domain. This technique can also be used in the Twitter environment. Specifically, for the scenario of this research, it is possible to examine the domain ontology to discover the relationship and lexical variation of the terms in tweets once the event is defined by the user. After that, rather than considering each term in the document as equally important, terms that co-exists in that domain ontology can be higher weighted. Also, new terms which do not exist in the tweet corpus but belong to the same domain ontology can be added to the text model.

On the other hand, the proposed adaptive crawling model can also use to enrich the domain ontologies. Existing research [88] demonstrates that the mechanism in the proposed adaptive crawling model is capable of supporting event lexicon building. As a result, a prospective direction is to combine the aforementioned two processes together to develop a mutual reinforcement model so to improve the accuracy for event content identification.

- The proposed adaptive crawling is triggered by a set of pre-defined keywords (in section 3.2). Although it is possible to apply the adaptive crawling to any events once the event theme is known, the requirement on knowing a priori event-related initial keywords restricts its utilisation for unplanned events. A manually triggered process risks losing time-sensitive information that is often critical for unplanned event information acquisition and dissemination. As a result, one of the adaptive crawling refinements lies in the automatic identification of the event theme. This can be considered as an extension of the work already done on the existing FSD or NED detection (in section 2.2.2). After identifying the first story from a FSD or NED algorithm, the key problems are how to automatically recognise the importance of the event and then to synthesis out of the available 'raw' information, a concise but concrete summarisation within a limited amount of time.
- Identification of noisy tweets itself is not an easy task due to the informal usage and short length of tweets. However, as shown in section 4.3.1, though the current version of KwAA tries to minimise the noise while retaining the coverage of the event traffic, it still incurs unexpected results for the event detection task. The amount of noise in the Twitterverse is too notable to be ignored by the existing real-time event detection algorithms. Exploiting techniques used in tweets retrieval systems using more sophisticated query generation or automatic tweets classification can be candidate solutions for distinguishing the noisy tweets in a timely fashion.

2. **Event tracking and profiling with new event detection algorithm that can incorporate the adaptive crawling** (extension of chapter 4).

This research shows that adaptive crawling reveals useful information concerning evolution or unfolding of an event by detecting more realistic sub-events for planned event. However, tweets about different, overlapping sub-events are often mixed up with each other and cannot be directly distinguished by the existing event detection algorithms. Updating the exiting event detection algorithm to track important sub-events and profile them individually will not only improve the event awareness but also facilitate more accurate keywords adaptation. For example, when collecting tweets about the Ukraine crisis[1], the adaptive crawler identifies tweets about the crashing of the Malaysia Airlines Flight 17 (MH17)[2] a priori to the baseline and keeps track of the tweets about it. Providing an event detection algorithm that can detect, track and profile the MH17 sub-event directly from the Ukraine crisis tweets stream, it is possible to achieve a quicker and more comprehensive knowledge about the MH17 event using online social networks.

3. **Investigation of the effects of adaptive crawling on other event monitoring applications**

   Compared with traditional newswires, social media services provide easier access, enabling the general public to express their opinions and judgements about real-world events. Twitter, as the most preferred social media service for breaking news [28] and entity-oriented topics [130], accumulates people's opinions and sentiments about the social and news events [131]. The online discussions about particular events thus provide opportunities for event monitoring through opinion mining and sentiment analysis [111, 132]. As a result, a further future direction is to extend this work to investigate the effects of the proposed adaptive crawling on sentiment analysis, such as the coverage of the opinions and the propagation of sentiment.

[1] This is a political movement between Ukraine, European Union and Russia, more details in `https://en.wikipedia.org/wiki/Ukrainian_crisis`

[2] A plane belongs to civil aviation company that are wrongly shot down by military: `https://en.wikipedia.org/wiki/Malaysia_Airlines_Flight_17`

Appendix A

Datasets Overview of Crawled Events

Table A.1: Datasets Overview

Event	Keywords (Initial Seeds)	Collection period and Corpus size			
		Baseline	TF-KwAA	TP-KwAA	CS-KwAA
2012 London Olympic Games	Olympic, #London2012	2012-07-27, 20:39 to 2012-08-28, 08:30 (18465672)	2012-07-27, 20:41 to 2012-08-11, 17:52 (58759453)	-	-
2013 Wimbledon Championships	Wimbledon, #wimbledon2013	2013-06-24, 21:41 to 2013-06-28, 15:54 (861641)	2013-06-24, 21:41 to 2013-06-27, 18:37 (11539738)	2013-06-24, 21:41 to 2013-06-28, 16:10 (1767146)	2013-06-24, 21:41 to 2013-06-28, 16:03 (1049684)
2013 Glastonbury Music Festival	Glastonbury	2013-06-28, 16:26 to 2013-07-02, 06:52 (643612)	2013-06-28, 16:26 to 2013-07-02, 09:22 (15418924)	22013-06-28, 16:26 to 2013-07-02, 10:22 (4325347)	2013-06-28, 16:26 to 2013-07-02, 10:22 (898101)
2013 Egypt Protest	Egypt protest, #ArabSpring	2013-07-17, 21:19 to 2013-07-18, 16:41 (77277)	2013-07-17, 21:19 to 2013-07-18, 16:40 (2887165)	2013-07-17, 21:19 to 2013-07-18, 16:41 (719931)	2013-07-17, 21:19 to 2013-07-18, 16:44 (219911)
Missing Plane Malaysia Airlines Flight 370 (MH370)	Malaysia Airlines, MH370	2014-03-09, 22:21 to 2014-06-09, 15:35 (11826943)	-	2014-03-20, 11:13 to 2014-03-20, 21:55 (1448409)	2014-03-09, 22:17 to 2014-06-04, 19:37 (22944013)

TABLE A.2: Datasets Overview (Continued)

Event	Keywords (Initial Seeds)	Collection period and Corpus size			
		Baseline	TF-KwAA	TP-KwAA	CS-KwAA
Philippine Earthquake	Philippines earthquake, #earthquake	2013-10-15, 17:14 to 2013-11-04, 11:18 (950234)	-	-	2013-10-15, 17:14 to 2013-11-04, 11:15 (3369905)
2014 Sochi Winter Olympic Games	sochi, #olympic2014, #sochi2014	2014-02-10, 15:41 to 2014-02-27, 12:37 (6357977)	-	-	2014-02-10, 15:41 to 2014-02-27, 14:00 (9421847)
Ukraine Crisis	#Ukraine	2014-03-03, 10:12 to 2014-08-27, 12:44 (6112965)	-	-	2014-03-03, 10:12 to 2014-08-23, 11:48 (14947891)
Malaysia Airlines Flight 17 (MH17)	MH17	2014-07-21, 17:07 to 2014-08-27, 09:05 (3325070)	-	-	2014-07-21, 17:04 to 2014-08-23, 11:50 (4475629)
2014 World Cup	world cup, #worldcup	2014-06-09, 20:37 to 2014-07-17, 11:26 (51942513)	-	-	2014-06-06, 20:35 to 2014-07-13, 20:02 (76787570)
2014 Scottish Referendum	scottish referendum, #Scottish-Referendum	2014-09-10, 18:01 to 2014-09-23, 18:08 (360659)	-	-	2014-09-10, 11:41 to 2014-09-22, 10:42 (4109325)

Appendix B

Event Detection Results

This Appendix gives an example output for the detection results that are produced in chapter by 4. The statistical based burst detection algorithm which is proposed in Twitinfo event monitoring system is applied.

As shown in Table B.1, detected peak window, summary tweets and event entropy are listed. This is based on filtered datasets of Glastonbury Festival.

In this table, the sub-events are referred by their indexes. The relationships between sub-event indexes and their title are listed in Table 4.6.

Table B.1: Detailed Event Detection Results (filtered BL dataset for Glastonbury)

window Idx	peak window	summary tweet	sub-event	entropy
1	2013-06-29, 11:30:00, BST to 2013-06-29, 11:39:59, BST	At Glastonbury with sunshine, bacon and 3G. I"m pinching myself		
2	2013-06-29, 12:50:00, BST to 2013-06-29, 13:24:59, BST	Zoek mij Glastonbury Festival... Gister 135.000 visitors `http://t.co/XE1O9Wzb9b` Succes!		
3	2013-06-29, 13:25:00, BST to 2013-06-29, 14:04:59, BST	An arial view of Glastonbury Festival, England. 135.000 visitors `http://t.co/GjNv9n7TD9`		
4	2013-06-29, 15:40:00, BST to 2013-06-29, 15:54:59, BST	BEN HOWARD AT GLASTONBURY uahhhhhh wish i'd gone	2	25.37
5	2013-06-29, 15:55:00, BST to 2013-06-29, 16:19:59, BST	Laura Mvula looks so boring at Glastonbury	3	21.98
6	2013-06-29, 16:20:00, BST to 2013-06-29, 17:19:59, BST	Ben howard is on bbc2 right now #glastonbury EmilyJFairhurst	2	29.82
7	2013-06-29, 17:20:00, BST to 2013-06-29, 18:04:59, BST	They"re interviewing a woman at Glastonbury who is a HULA HOOP THERAPIST	4	18.98
8	2013-06-29, 19:30:00, BST to 2013-06-29, 19:44:59, BST	Noah & the whale baby!! #glastonbury	5	24.39
9	2013-06-29, 19:45:00, BST to 2013-06-29, 19:54:59, BST	Noah And The Whale at Glastonbury, fantastic!!	5	25.93
10	2013-06-29, 19:55:00, BST to 2013-06-29, 21:14:59, BST	#glastonbury badger badger badger badger badger badger	9	34.54
11	2013-06-29, 21:15:00, BST to 2013-06-29, 21:39:59, BST	Wonder will Eavis get "BADGERED" tomorrow at #Glastonbury? - "BADGER BADGER BADGER!"	9	23.36
12	2013-06-29, 21:40:00, BST to 2013-06-29, 22:44:59, BST	I think I"d be quite into Glastonbury if I was some kind of predator/ serial killer		
13	2013-06-29, 22:45:00, BST to 2013-06-29, 23:24:59, BST	Oh dear the #Stones at #glastonbury look like a Wonga TV ad!!	11	29.00

Bibliography

[1] James Allan. Introduction to topic detection and tracking. In *Topic detection and tracking*, pages 1–16. Springer, 2002.

[2] Yiming Yang, Tom Pierce, and Jaime Carbonell. A study of retrospective and online event detection. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 28–36, New York, NY, USA, 1998. ACM.

[3] James Allan, Jaime G Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. Topic detection and tracking pilot study final report. 1998.

[4] David Niven. Bias in the news: Partisanship and negativity in media coverage of presidents george bush and bill clinton. *The Harvard International Journal of Press/Politics*, 6(3):31–46, 2001.

[5] Frank Barnas and Ted White. *Broadcast News: Writing, Reporting, and Producing.* Focal Press, 2013.

[6] Tim O'Reilly. What is web 2.0: Design patterns and business models for the next generation of software. 2005.

[7] Michael Schudson. *The Sociology of News.* Contemporary societies. Norton, 2003.

[8] Nic Newman, David Levy, and Kleis Nielsen. Digital news report 2015 [online]. Technical report, Reuters Institute for the Study of Journalism, `http://www.digitalnewsreport.org/`, 2015.

[9] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 591–600, New York, NY, USA, 2010. ACM.

[10] Siqi Zhao, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. Human as real-time sensors of social and physical events: A case study of twitter and sports games. *CoRR*, abs/1106.4300, 2011.

[11] Dhiraj Murthy. Twitter: Microphone for the masses. *Media Culture Society*, 33(5):779–789, 2011.

[12] Leysia Palen, Sarah Vieweg, Sophia B. Liu, and Amanda Lee Hughes. Crisis in a networked world. *Social Science Computer Review*, 27(4):467–480, November 2009.

[13] Onook Oh, Manish Agrawal, and H Raghav Rao. Information control and terrorism: Tracking the mumbai terrorist attack through twitter. *Information Systems Frontiers*, 13(1):33–43, 2011.

[14] Gilad Lotan, Erhardt Graeff, Mike Ananny, Devin Gaffney, Ian Pearce, and Danah Boyd. The arab spring— the revolutions were tweeted: Information flows during the 2011 tunisian and egyptian revolutions. *International Journal of Communication*, 5(0), 2011.

[15] Philip N. Howard, Aiden Duffy, Deen Freelon, Muzammil Hussain, Will Mari, and Marwa Mazaid. Opening closed regimes: What was the role of social media during the arab spring? 2011.

[16] Robin Stephenson and Peter S. Anderson. Disasters and the information technology revolution. *Disasters*, 21(4):305–334, 1997.

[17] Marr. Bernard. Can big data algorithms tell better stories than humans? [online]. `http://www.datasciencecentral.com/profiles/blogs/can-big-data-algorithms-tell-better-stories-than-humans`, September 3, 2015.

[18] Qiankun Zhao, Prasenjit Mitra, and Bi Chen. Temporal and information flow based event detection from social text streams. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 1501–1506, 2007.

[19] Hassan Sayyadi, Matthew Hurst, and Alexey Maykov. Event detection and tracking in social streams. In *Proceedings of the Third International Conference on Weblogs and Social Media, ICWSM 2009, San Jose, California, USA, May 17-20, 2009*, 2009.

[20] Fotis Psallidas, Hila Becker, Mor Naaman, and Luis Gravano. Effective event identification in social media. *IEEE Data Eng. Bull.*, 36(3):42–50, 2013.

[21] Dhiraj Murthy. *Twitter: Social Communication in the Twitter Age*. Polity Press, 2013.

[22] Matthias Revers. The twitterization of news making: Transparency and journalistic professionalism. *Journal of Communication*, 64(5):806–826, 2014.

[23] Twitter Inc. Twitter reports: Second quarter 2015[online]. Technical report, Twitter Inc., `https://investor.twitterinc.com/results.cfm`, July 28, 2015.

[24] Aaron Moy. Four insights about millennials on twitter [online]. `https://blog.twitter.com/2014/four-insights-about-millennials-on-twitter`, July 9, 2014.

[25] David A. Shamma, L. Kennedy, and Elizabeth F. Churchill. Tweetgeist: Can the twitter timeline reveal the structure of broadcast events? *CSCW Horizons*, 2010.

[26] Mitsumasa Kubo, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. Generating live sports updates from twitter by finding good reporters. In *2013 IEEE/WIC/ACM International Conferences on Web Intelligence, WI 2013, Atlanta, GA, USA, November 17-20*, pages 527–534, 2013.

[27] Miles Osborne and Mark Dredze. Facebook, twitter and google plus for breaking news: Is there a winner? In *Proceedings of the Eighth International Conference*

on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4., 2014.

[28] Saša Petrović, Miles Osborne, Richard McCreadie, Craig Macdonald, Iadh Ounis, and Luke Shrimpton. Can twitter replace newswire for breaking news? In *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM 2013, Cambridge, Massachusetts, USA, July 8-11,*, 2013.

[29] Jonathan Hurlock and Max L. Wilson. Searching twitter: Separating the tweet from the chaff. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21*, 2011.

[30] Fabricio Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. Detecting spammers on twitter. In *In Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS*, 2010.

[31] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1,*, pages 675–684, 2011.

[32] Saša Petrović, Miles Osborne, and Victor Lavrenko. Streaming first story detection with application to twitter. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 2-4, 2010, Los Angeles, California, USA*, pages 181–189, 2010.

[33] Jianshu Weng and Bu-Sung Lee. Event detection in twitter. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*, 2011.

[34] Hila Becker, Mor Naaman, and Luis Gravano. Beyond trending topics: Real-world event identification on twitter. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*, 2011.

[35] Ana-Maria Popescu, Marco Pennacchiotti, and Deepa Paranjpe. Extracting events and event descriptions from twitter. In *Proceedings of the 20th International Con-*

ference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011 (Companion Volume), pages 105–106, 2011.

[36] Chenliang Li, Aixin Sun, and Anwitaman Datta. Twevent: segment-based event detection from tweets. In *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02*, pages 155–164, 2012.

[37] Alan Ritter, Mausam, Oren Etzioni, and Sam Clark. Open domain event extraction from twitter. In *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012*, pages 1104–1112, 2012.

[38] Wei Xie, Feida Zhu, Jing Jiang, Ee-Peng Lim, and Ke Wang. Topicsketch: Real-time bursty topic detection from twitter. In *2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, December 7-10, 2013*, pages 837–846, 2013.

[39] Alexandra Olteanu, Carlos Castillo, Nicholas Diakopoulos, and Karl Aberer. Comparing events coverage in online news and social media: The case of climate change. In *Proceedings of the Ninth International Conference on Web and Social Media, ICWSM 2015, University of Oxford, Oxford, UK, May 26-29,*, pages 288–297, 2015.

[40] Tom Rosenstiel, Jeff Sonderman, Kevin Loker, Maria Ivancin, and Nina Kjarval. Twitter and breaking news [online]. `http://www.americanpressinstitute.org/publications/reports/survey-research/twitter-and-breaking-news/`, Jan 9, 2015.

[41] Jie Yin, Andrew Lampert, Mark Cameron, Bella Robinson, and Robert Power. Using social media to enhance emergency situation awareness. *IEEE Intelligent Systems*, (6):52–59, 2012.

[42] Daniela Pohl, Abdelhamid Bouchachia, and Hermann Hellwagner. Automatic sub-event detection in emergency management using social media. In *Proceedings of*

the 21st World Wide Web Conference, WWW 2012, Lyon, France, April 16-20, 2012 (Companion Volume), pages 683–686, 2012.

[43] Adam Marcus, Michael S Bernstein, Osama Badar, David R Karger, Samuel Madden, and Robert C Miller. Twitinfo: aggregating and visualizing microblogs for event exploration. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 227–236. ACM, 2011.

[44] Dhekar Abhik and Durga Toshniwal. Sub-event detection during natural hazards using features of social media data. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013, Companion Volume*, pages 783–788, 2013.

[45] Flavio Chierichetti, Jon M. Kleinberg, Ravi Kumar, Mohammad Mahdian, and Sandeep Pandey. Event detection via communication pattern analysis. In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014.*, 2014.

[46] Carlos Martin, David Corney, and Ayse Göker. Mining newsworthy topics from social media. In *Advances in Social Media Analysis*, pages 21–43. 2015.

[47] Polykarpos Meladianos, Giannis Nikolentzos, François Rousseau, Yannis Stavrakas, and Michalis Vazirgiannis. Degeneracy-based real-time sub-event detection in twitter stream. In *Proceedings of the Ninth International Conference on Web and Social Media, ICWSM 2015, University of Oxford, Oxford, UK, May 26-29, 2015*, pages 248–257, 2015.

[48] Chao Shen, Fei Liu, Fuliang Weng, and Tao Li. A participant-based approach for event summarization using twitter streams. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 1152–1162, 2013.

[49] Hila Becker, Dan Iter, Mor Naaman, and Luis Gravano. Identifying content for planned events across social media sites. In *Proceedings of the Fifth ACM Inter-*

national Conference on Web Search and Data Mining, WSDM '12, pages 533–542, New York, NY, USA, 2012. ACM.

[50] Heather S. Packer, Sina Samangooei, Jonathon S. Hare, Nicholas Gibbins, and Paul H. Lewis. Event detection using twitter and structured semantic query expansion. In *Proceedings of the 1st International Workshop on Multimodal Crowd Sensing*, CrowdSens '12, pages 7–14, New York, NY, USA, 2012. ACM.

[51] Sayan Unankard, Xue Li, Mohamed A. Sharaf, Jiang Zhong, and Xueming Li. Predicting elections from social networks based on sub-event detection and sentiment analysis. In *Web Information Systems Engineering - WISE 2014 - 15th International Conference, Thessaloniki, Greece, October 12-14, 2014, Proceedings, Part II*, pages 1–16, 2014.

[52] Luca Maria Aiello, Georgios Petkos, Carlos J. Martín, David Corney, Symeon Papadopoulos, Ryan Skraba, Ayse Göker, Ioannis Kompatsiaris, and Alejandro Jaimes. Sensing trending topics in twitter. *IEEE Transactions on Multimedia*, 15(6):1268–1282, 2013.

[53] Georgiana Ifrim, Bichen Shi, and Igor Brigadir. Event detection in twitter using aggressive filtering and hierarchical tweet clustering. In *Proceedings of the SNOW 2014 Data Challenge co-located with 23rd International World Wide Web Conference (WWW 2014), Seoul, Korea, April 8, 2014.*, pages 33–40, 2014.

[54] Daniela Pohl, Abdelhamid Bouchachia, and Hermann Hellwagner. Supporting crisis management via detection of sub-events in social networks. *International Journal of Information Systems for Crisis Response and Management (IJISCRAM)*, 5(3):20–36, 2013.

[55] Zellig Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

[56] David M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4):77–84, April 2012.

[57] Raffi Krikorian. New tweets per second record, and how! [online]. `https://blog.twitter.com/2013/new-tweets-per-second-record-and-how`, 2013.

[58] Saša Petrović. *Real-time Event Detection in Massive Streams*. PhD thesis, University of Edinburgh, 2012.

[59] Ramesh Nallapati. Discriminative models for information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 64–71, New York, NY, USA, 2004. ACM.

[60] Y. Demchenko, Zhiming Zhao, P. Grosso, A. Wibisono, and C. de Laat. Addressing big data challenges for scientific data infrastructure. In *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*, pages 614–617, Dec 2012.

[61] Yiming Yang, Jaime G Carbonell, Ralf D Brown, Thomas Pierce, Brian T Archibald, and Xin Liu. Learning approaches for detecting and tracking news events. *IEEE Intelligent Systems*, (4):32–43, 1999.

[62] Elena Filatova, Vasileios Hatzivassiloglou, and Kathleen McKeown. Automatic creation of domain templates. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 207–214. Association for Computational Linguistics, 2006.

[63] Jonathan G Fiscus and George R Doddington. Topic detection and tracking evaluation overview. *Topic detection and tracking*, pages 17–31, 2002.

[64] April Kontostathis, Leon M Galitsky, William M Pottenger, Soma Roy, and Daniel J Phelps. A survey of emerging trend detection in textual data mining. In *Survey of Text Mining*, pages 185–224. Springer, 2004.

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

[66] Xuanhui Wang, ChengXiang Zhai, Xiao Hu, and Richard Sproat. Mining correlated bursty topic patterns from coordinated text streams. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 784–793. ACM, 2007.

[67] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.

[68] Manish Gupta, Jing Gao, Charu C. Aggarwal, and Jiawei Han. Outlier detection for temporal data: A survey. *IEEE Trans. Knowl. Data Eng.*, 26(9):2250–2267, 2014.

[69] Daniel M Romero, Brendan Meeder, and Jon Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 695–704. ACM, 2011.

[70] Hila Becker. *Identification and Characterization of Events in Social Media.* PhD thesis, Columbia University, 2011.

[71] Mor Naaman, Hila Becker, and Luis Gravano. Hip and trendy: Characterizing emerging trends on twitter. *Journal of the American Society for Information Science and Technology*, 62(5):902–918, 2011.

[72] Anqi Cui, Min Zhang, Yiqun Liu, Shaoping Ma, and Kuo Zhang. Discover breaking events with popular hashtags in twitter. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1794–1798. ACM, 2012.

[73] Martijn Spitters and Wessel Kraaij. A language modeling approach to tracking news events. In *In Proceedings of TDT workshop 2000.* Citeseer, 2000.

[74] James Allan, Victor Lavrenko, and Russell Swan. Explorations within topic tracking and detection. In *Topic detection and tracking*, pages 197–224. Springer, 2002.

[75] Farzindar Atefeh and Wael Khreich. A survey of techniques for event detection in twitter. *Comput. Intell.*, 31(1):132–164, February 2015.

[76] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why we twitter: Understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network*

Analysis, WebKDD/SNA-KDD '07, pages 56–65, New York, NY, USA, 2007. ACM.

[77] Raffi Krikorian. Map of a twitter status object [online]. `https://dev.twitter.com/overview/api/tweets`, 2010.

[78] Twitter. Tracking parameter for streaming api [online]. `https://dev.twitter.com/streaming/overview/request-parameters#track`, 2012.

[79] Kenneth Joseph, Peter M. Landwehr, and Kathleen M. Carley. Two 1%s don't make a whole: Comparing simultaneous samples from twitter's streaming API. In *Social Computing, Behavioral-Cultural Modeling and Prediction - 7th International Conference, SBP 2014, Washington, DC, USA, April 1-4, 2014. Proceedings*, pages 75–83, 2014.

[80] Fred Morstatter, Jürgen Pfeffer, Huan Liu, and Kathleen M. Carley. Is the sample good enough? comparing data from twitter's streaming API with twitter's firehose. In *Proceedings of the Seventh International Conference on Weblogs and Social Media*, ICWSM '13, 2013.

[81] Joao Gama. *Knowledge discovery from data streams*. CRC Press, 2010.

[82] Ao Feng and James Allan. Hierarchical topic detection in tdt-2004. *Center for Intelligent Information Retrieval. University of Massachusetts, Amherst*, 2005.

[83] Albert Bifet and Richard Kirkby. Data stream mining a practical approach, 2009.

[84] Sasa Petrovic, Miles Osborne, and Victor Lavrenko. The edinburgh twitter corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, pages 25–26, 2010.

[85] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 851–860, New York, NY, USA, 2010. ACM.

[86] Kate Starbird and Leysia Palen. (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*, CSCW '12, pages 7–16, 2012.

[87] Jeffrey Nichols, Jalal Mahmud, and Clemens Drews. Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*, IUI '12, pages 189–198, 2012.

[88] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014.*, 2014.

[89] Meeyoung Cha, Hamed Haddadi, Fabricio Benevenuto, and P Krishna Gummadi. Measuring user influence in twitter: The million follower fallacy. *ICWSM*, 10(10-17):30, 2010.

[90] Changhyun Byun, Yanggon Kim, Hyeoncheol Lee, and Kwangmi Ko Kim. Automated twitter data collecting tool and case study with rule-based analysis. In *Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services*, pages 196–204. ACM, 2012.

[91] Matko Boanjak, Eduardo Oliveira, José Martins, Eduarda Mendes Rodrigues, and Luís Sarmento. Twitterecho: a distributed focused crawler to support open research with twitter data. In *Proceedings of the 21st international conference companion on World Wide Web*, pages 1233–1240. ACM, 2012.

[92] Pattisapu Nikhil Priyatam, Ajay Dubey, Krish Perumal, Sai Praneeth, Dharmesh Kakadia, and Vasudeva Varma. Seed selection for domain-specific search. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 923–928. International World Wide Web Conferences Steering Committee, 2014.

[93] Hila Becker, Mor Naaman, and Luis Gravano. Selecting quality twitter content for events. *ICWSM*, 11, 2011.

[94] Fabian Abel, Ilknur Celik, Geert-Jan Houben, and Patrick Siehndel. Leveraging the semantics of tweets for adaptive faceted search on twitter. In *Proceedings of the 10th International Conference on The Semantic Web - Volume Part I*, ISWC'11, pages 1–17, Berlin, Heidelberg, 2011. Springer-Verlag.

[95] Yajuan Duan, Long Jiang, Tao Qin, Ming Zhou, and Heung-Yeung Shum. An empirical study on learning to rank of tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 295–303. Association for Computational Linguistics, 2010.

[96] Xin Zhang, Ben He, Tiejian Luo, and Baobin Li. Query-biased learning to rank for real-time twitter search. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1915–1919. ACM, 2012.

[97] James Lanagan and Alan F. Smeaton. Using twitter to detect and tag important events in sports media. In *Proceedings of the Fifth International Conference on Weblogs and Social*, ICWSM '11, 2011.

[98] Nut Limsopatham, M-Dyaa Albakour, Craig Macdonald, and Iadh Ounis. Tweeting behaviour during train disruptions within a city. In *Proceedings of Digital Placemaking: Augmenting Physical Places with Contextual Social Data workshop at ICWSM*, Oxford, United Kingdom, 2015. AAAI.

[99] Shamanth Kumar, Huan Liu, Sameep Mehta, and L. Venkata Subramaniam. From tweets to events: Exploring a scalable solution for twitter streams. *CoRR*, abs/1405.1392, 2014.

[100] Richard McCreadie, Craig Macdonald, Iadh Ounis, Miles Osborne, and Sasa Petrovic. Scalable distributed event detection for twitter. In *Proceedings of the 2013 IEEE International Conference on Big Data, 6-9 October 2013, Santa Clara, CA, USA*, pages 543–549, 2013.

[101] Ana-Maria Popescu and Marco Pennacchiotti. Detecting controversial events from twitter. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010*, pages 1873–1876, 2010.

[102] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, September 1999.

[103] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158. ACM, 2010.

[104] Paul S Earle, Daniel C Bowden, and Michelle Guy. Twitter earthquake detection: earthquake monitoring in a social world. *Annals of Geophysics*, 54(6), 2012.

[105] Sophia B. Liu and Leysia Palen. Spatiotemporal mashups: A survey of current tools to inform next generation crisis support. In *In Proceedings of 6th International Conference on Information Systems for Crisis Response and Management: Boundary Spanning Initiatives and New Perspectives*, ISCRAM 2009, 2009.

[106] Franz Wanner, Andreas Stoffel, Dominik Jäckle, BC Kwon, Andreas Weiler, Daniel A Keim, Katherine E Isaacs, Alfredo Giménez, Ilir Jusufi, Todd Gamblin, et al. State-of-the-art report of visual analysis for event detection in text data streams. In *Computer Graphics Forum*, volume 33, 2014.

[107] Wei Xu, Ralph Grishman, Adam Meyers, and Alan Ritter. A preliminary study of tweet summarization using information extraction. *NAACL 2013*, page 20, 2013.

[108] David Inouye and Jugal K. Kalita. Comparing twitter summarization algorithms for multiple post summaries. In *PASSAT/SocialCom 2011, Privacy, Security, Risk and Trust (PASSAT), 2011 IEEE Third International Conference on and 2011 IEEE Third International Conference on Social Computing (SocialCom), Boston, MA, USA, 9-11 Oct., 2011*, pages 298–306, 2011.

[109] Richard McCreadie, Ian Soboroff, Jimmy Lin, Craig Macdonald, Iadh Ounis, and Dean McCullough. On building a reusable twitter corpus. In *Proceedings of the*

35th international ACM SIGIR conference on Research and development in information retrieval, pages 1113–1114. ACM, 2012.

[110] Andrew J McMinn, Yashar Moshfeghi, and Joemon M Jose. Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 409–418. ACM, 2013.

[111] Andranik Tumasjan, Timm Oliver Sprenger, Philipp G. Sandner, and Isabell M. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International Conference on Weblogs and Social Media*, ICWSM '10, 2010.

[112] Muhammad Bilal Zafar, Parantapa Bhattacharya, Niloy Ganguly, Krishna P. Gummadi, and Saptarshi Ghosh. Sampling content from online social networks: Comparing random vs. expert sampling of the twitter stream. *ACM Trans. Web*, 9(3):12:1–12:33, June 2015.

[113] Oren Tsur and Ari Rappoport. What's in a hashtag?: Content based prediction of the spread of ideas in microblogging communities. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, WSDM '12, pages 643–652, 2012.

[114] Deepayan Chakrabarti and Kunal Punera. Event summarization using tweets. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, (ICWSM'11) Barcelona, Catalonia, Spain, July 17-21, 2011*, 2011.

[115] Albert Bifet, Geoffrey Holmes, and Bernhard Pfahringer. Moa-tweetreader: Real-time analysis in twitter streaming data. In *Discovery Science - 14th International Conference, DS 2011, Espoo, Finland, October 5-7, 2011. Proceedings*, pages 46–60, 2011.

[116] Andrea Varga, Amparo Elizabeth Cano, and Fabio Ciravegna. Exploring the similarity between social knowledge sources and twitter for cross-domain topic

classification. Knowledge Extraction and Consolidation from Social Media (KECSM2012), International Semantic Web Conference 2012 (ISWC 2012), 2012.

[117] Fernando Perez-Tellez, David Pinto, John Cardiff, and Paolo Rosso. On the difficulty of clustering company tweets. In *Proceedings of the 2Nd International Workshop on Search and Mining User-generated Contents*, SMUC '10, pages 95–102, 2010.

[118] Rishabh Mehrotra, Scott Sanner, Wray L. Buntine, and Lexing Xie. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013*, pages 889–892, 2013.

[119] Anna Huang. Similarity measures for text document clustering. In *Proceedings of the Sixth New Zealand Computer Science research student Conference*, NZCSRSC 2008, pages 49–56, 2008.

[120] Cyril Cleverdon. The cranfield tests on index language devices. *Aslib Proceedings of Information Management*, 19(6):173–194, 1967.

[121] Shiva Imani Moghadasi, Sri Devi Ravana, and Sudharshan N. Raman. Low-cost evaluation techniques for information retrieval systems: A review. *J. Informetrics*, 7(2):301–312, 2013.

[122] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[123] Karl Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58(347-352):240–242, 1895.

[124] Brian Everitt. *The Cambridge dictionary of statistics*. Cambridge University Press, Cambridge, UK; New York, 2002.

[125] Xinyue Wang, Laurissa Tokarchuk, Félix Cuadrado, and Stefan Poslad. Exploiting hashtags for adaptive microblog crawling. In *Advances in Social Networks Analysis*

and Mining 2013, ASONAM '13, Niagara, ON, Canada - August 25 - 29, 2013, pages 311–315, 2013.

[126] V Paxson and M Allman. Rfc 2988-computing tcp's re-transmission timer (november 2000). Technical report, Internet RFC.

[127] Ke Tao, Fabian Abel, Claudia Hauff, and Geert-Jan Houben. What makes a tweet relevant for a topic. In *Proceedings of the Making Sense of Microposts (# MSM2012) Workshop*, pages 49–56. CEUR, 2012.

[128] C. E. Shannon. A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(1):3–55, January 2001.

[129] J. Bhogal, A. Macfarlane, and P. Smith. A review of ontology based query expansion. *Information Processing & Management*, 43(4):866 – 886, 2007.

[130] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. In *Proceedings of Advances in Information Retrieval - 33rd European Conference on IR Research, ECIR 2011, Dublin, Ireland, April 18-21,*, pages 338–349, 2011.

[131] Alexander Pak and Patrick Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, Valletta, Malta, May 17-23,*, 2010.

[132] Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. Sentiment in twitter events. *J. Am. Soc. Inf. Sci. Technol.*, 62(2):406–418, February 2011.