TweetSpy: A Machine Learning Approach to Detecting Political Unrest Using Twitter

By
Haroon Raja
2009-NUST-MS-EE-08

Supervisor
Dr. Muhammad Usman Ilyas
NUST-SEECS

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Electrical Engineering (MS EE)

At the
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(March 2012)

Approval

It is certified that the contents and form of thesis entitled "**TweetSpy: A Machine Learning Approach to Detecting Political Unrest Using Twitter**" submitted by **Haroon Raja** have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Muhammad Usman Ilyas**

Signature: _____
Date: _____

Committee Member: **Dr. Aimal Rextin**

Signature: _____
Date: _____

Committee Member: **Dr. Syed Ali Khayyam**

Signature: _____
Date: _____

Committee Member: **Dr. Zahid Anwar**

Signature: _____
Date: _____

Dedication

To my parents.

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECS or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Haroon Raja**

Signature: _____

Acknowledgements

I would like to thank all the individuals who have helped me directly or indirectly for my thesis work. In specific I am very grateful to my adviser Dr Usman Ilyas who has been the key figure in timely completion of my thesis. The time I joined Dr Usman for my thesis work I was too much short of confidence and self belief regarding my aptitude as a researcher, working on thesis problem and appreciation by Dr Usman has provided me with immense confidence. In short I personally believe I have grown as a researcher during this time. I would also like to thank our collaborators at MSU, Dr Alex Liu, Dr Hayder Radha and Zubair Shafique for their guidance through the course of thesis work. At the start I learned a lot from Zubair about parsing useful information from the data set, I am extremely thankful to him for this.
At SEECS my committee members, Dr Aimal Rextin, Dr Ali Khayam and Dr Zahid Anwar were very helpful throughout my thesis. Their comments at the proposal defense helped us to re-define some of our assumptions which eventually led to a stronger thesis work. I would also like to mention the faculty members for teaching us good stuff which ultimately helped in the thesis research work. I would especially like to mention Dr Aimal's effort as a graph theory instructor, he taught the course in unconventional manner which made the course a lot more interesting and has helped me great deal in my thesis. Among the instructors for my Masters course work I feel thoroughly indebted to Dr Ali Khayam, his way of teaching and his research vision has helped all along this strenuous path, right from the start to the completion of MS degree. I would also like to thank my fellow students at SEECS; Ali, Hassan, Jaweria, Waqar and Shahzad for the research discussions and for making the time spent at SEECS the memorable one. I am thankful to my friends Hussain Kazmi and Sarmad Munir for soft banter about completion of my thesis time and again, this acted as a constant reminder and helped me to stay focused.
Lastly, I feel proud the way my parents and younger siblings have shown patience and confidence in my abilities. And they were always there to provide

me with their full support in the difficult of times. This thesis may not have been possible without the sacrifices made by them.

Abstract

The popular uprisings in a number of countries in the Middle East and North Africa in the Spring of 2011 were enabled in large part by local populations access to social networking services such as Twitter and Facebook. This thesis attempts to use language independent features of Twitter traffic mentioning different countries to distinguish between countries that are politically unstable and others that are stable. Towards this end, we collected several data sets of countries that were experiencing political unrest during the period now known as the Arab Spring, as well as a set of countries that were not. Several different methods are used to model the flow of information between Twitter users in data sets as graphs, called information cascades. Naïve Bayesian, Support Vector Machines (SVM) and Bayesian logistic regression classifiers are applied to all data sets. By using the dynamic properties of information cascades, Naïve Bayesian and SVM classifiers both achieve true positives rates of 100%, with false positives rates of 3% and 0%, respectively.

Table of Contents

1 Introduction **1**
- 1.1 Background and Motivation . . . 1
- 1.2 Limitations of Prior Art . . . 2
- 1.3 Proposed Solution . . . 2
- 1.4 Experimental Results and Findings . . . 3
- 1.5 Key Contributions . . . 4

2 Related Work **5**
- 2.1 Information Cascades . . . 5
- 2.2 Pattern Characterization . . . 6

3 Data Set **8**

4 Technical Approach **11**
- 4.1 Individual Tweet Analysis (M1) . . . 12
  - 4.1.1 Edge Count . . . 12
  - 4.1.2 Diffusion Lifetime . . . 12
- 4.2 Multiple Tweet Static Analysis (M2) . . . 13
  - 4.2.1 Node Degree . . . 14
  - 4.2.2 Node In-Degree . . . 14
  - 4.2.3 Node Out-Degree . . . 14
  - 4.2.4 Closeness Centrality . . . 15
  - 4.2.5 Clique Count . . . 15
  - 4.2.6 Triangle Count . . . 15
- 4.3 Multiple Tweet Dynamic Analysis (M3) . . . 16
  - 4.3.1 Relative Inclusion . . . 16
  - 4.3.2 Jaccard Index . . . 16
  - 4.3.3 Graph Edit Distance . . . 17
- 4.4 Tweet Content Analysis (M4) . . . 17
- 4.5 Classification . . . 18

5 Classification **19**
5.1 Overview 19
5.2 Classifiers 20
5.2.1 Logistic Regression 20
5.2.2 Naïve Bayesian 21
5.2.3 Support Vector Machine (SVM) 21
5.3 Feature Ranking Methods 23
5.4 N-fold cross validation 24

6 Evaluation **25**
6.1 Individual Tweet Analysis (M1) 25
6.2 Multiple Tweet Static Analysis (M2) 26
6.3 Multiple Tweet Dynamic Analysis (M3) 27
6.4 Tweet Content Analysis (M4) 29

7 Conclusions **45**
7.1 Summary 45
7.2 Future Directions 46

List of Figures

6.1 Examples of feature computation from the distribution of properties in individual tweet analysis method (M1). 31
6.2 Feature computation from the distribution of properties in multiple tweet static analysis method (M2). 32
6.2 Feature computation from the distribution of properties in multiple tweet static analysis method (M2). 33
6.2 Feature computation from the distribution of properties in multiple tweet static analysis method (M2). 34
6.3 R-squared values for multiple tweets static graph properties distribution fitted to a linear equaion. 35
6.4 Feature distributions from the distribution of properties in multiple tweet static analysis method (M2). 36
6.4 Feature distributions from the distribution of properties in multiple tweet static analysis method (M2). 37
6.4 Feature distributions from the distribution of properties in multiple tweet static analysis method (M2). 38
6.5 Feature distributions from the distribution of properties in multiple tweet dynamic analysis method (M3). 39
6.5 Feature distributions from the distribution of properties in multiple tweet dynamic analysis method (M3). 40
6.5 Feature distributions from the distribution of properties in multiple tweet dynamic analysis method (M3). 41
6.6 Feature distributions from the distribution of properties in tweet content analysis method (M4). 42
6.6 Feature distributions from the distribution of properties in tweet content analysis method (M4). 43
6.6 Feature distributions from the distribution of properties in tweet content analysis method (M4). 44

List of Tables

3.1 Basic statistics of Twitter data sets. 10

6.1 Feature ranking for multiple tweet static analysis method (M2). 26
6.2 Feature ranking for multiple tweet dynamic analysis method (M3). 28
6.3 Classification results of all methods used in our study. M1 is the individual tweet analysis method, M2 is the multiple tweet static analysis method, M3 is the multiple tweet dynamic analysis method, and M4 is the tweet content analysis method. 30

Chapter 1

Introduction

1.1 Background and Motivation

The role played by social media and social network services during the uprisings in the Middle East (ME) and North Africa (NA), particularly in Tunisia and Egypt, during the period now known as the 'Arab Spring,' is well documented. Twitter [15] and Facebook were two key enabling tools that allowed the local populations to organize themselves. In the case of Tunisia it is now known that this organization and mobilization began as early as 2 weeks before the first street protests. These tools enabled people to share information about sniper locations, water distribution points and assembly locations for more protests [11]. Very notably, crowds were effective in squashing propaganda and misinformation that was introduced in social media by the Tunisian government.

In a larger context, what these recent events in the ME/NA have shown is that governments may choose to ignore the public's sentiment, but at their own peril. Monitoring social media has become this decade's equivalent of the Cold War era's spy-in-the-sky (satellite surveillance). On the positive side, even in situations when more traditional means of electronic communication (TV, radio, telephones/cellphones, e-mail *etc.*) are unavailable, social media can be leveraged very effectively for the quick collection, dissemination and propagation of critical information.

The two social network services that were most widely credited with enabling these revolutions in the ME/NA are Facebook and Twitter. Recent changes in Facebook's terms of use restrict researchers from crawling

Facebook profiles and collecting large sets of data. Therefore, we focus our attention on Twitter. Twitter is very open to data collection, to the extent that it encourages it. Furthermore, data sets like those from Twitter are rare in the sense that they are not anonymized or redacted. It allows access to almost all data associated with a tweet, including message contents.

1.2 Limitations of Prior Art

Recent years have seen an increased interest in the analysis and modeling of social networks. Earlier work consisted in large part of measurement studies of social networks. Wilson, Boe, Sala, Puttaswamy and Zhao [28] studied the correlation between social ties and network traffic. Recently, Romero, Meeder and Kleinberg [23] analyzed data collected from Twitter and detected differences in the spread pattern for hashtags belonging to different type of topics. They collected data over a 48 hour period and used it as a dynamic graph consisting of two 24 hour periods. In [13], Gong, Teng, Livne, Brunetti and Adamic studied dynamic graphs and the relationship between novelty of information flowing between two nodes in a social network and the conductance of the link between them.

These previous works analyzed differences in the information propagation across different topics. However, to the best of the authors' knowledge, no previous study has explored differences in the information flow on Twitter between data sets of the same domain.

1.3 Proposed Solution

In this work we develop a method for classifying country names appearing in Twitter traffic (tweets) as politically *stable* or *unstable*. We propose to make this classification based on properties of graphs underlying information flow between Twitter users. We propose a multi-step approach to the problem of classifying countries as politically stable or unstable.

The first step is to extract a graph from the collection of tweets collected from Twitter. These graphs capture the flow of information between Twitter users and are called *information cascades*. There are several definitions and existing approaches to extracting information cascades from Twitter data sets. We used three different approaches to generate information cascades

and extracted features from them for each Twitter data set. All three approaches distinguish themselves from prior definitions of information cascades on Twitter on the basis that besides using retweets, we include information about mentions and replies in the cascades.

The second step, feature extraction, requires us to measure a set of features of the cascades generated from each country's Twitter data. Since the Twitter data sets we have collected contain messaging (tweet) contents as well, it would have been possible to apply natural language processing for feaure extraction, to this end the main obstacle was the heterogeneous nature of language used in data sets related to different countries.

The third step is classification. Classification can be further sub-divided into feature selection, classification and verification.

1.4 Experimental Results and Findings

Three techniques are employed for data set classification. They include the Naïve Bayesian classifier, a Support Vector Machine classifier and Bayesian Logistic Regression.

We performed classification based on tweet content based features and observed detection rates for data sets of unstable countries of 87.5%, 37.5% and 37.5%, respectively, with corresponding false positive rates of 5.3%, 0% and 5.3%, respectively. Thus, the Naive Bayesian classifier outperforms the other two by large margins on these features.

However, the same classifiers fare significantly worse when applied to features of graphs of individual tweets and their retweets. The true positives rates of data sets of unstable countries were 50%, 0% and 0%, respectively, while false positives rates of all three classifiers were 0%.

Classification was also attempted on features of graphs constructed in a similar fashion to those constructed by Kwak *et al.* [20]. The true positives rates were 89.3%, 87.5% and 98.2%, respectively. However, these were coupled with significant false positives rates of 28.2%, 20.5% and 41%, respectively.

Finally, we perform classification on a set of features specific to dynamic graphs, *i.e.* graphs that include temporal information. All three classifiers have true positives rates of 100%. However, the false positives rate for Bayesian logistic regression is 100%, rendering it useless. Naive Bayesian

classifier and SVM perform better with false positives of only 3% and 0%, respectively.

1.5 Key Contributions

Our contributions in this paper are threefold:

1. Data Set Collection: We collected data sets consisting of Tweets containing names of different countries from within and outside the ME/NA region in the Spring of 2011.[1] To ensure the robustness of proposed detection algorithms we also collected data related to politics and movies as well.

2. Information Cascade Formation: We form information cascades modeling the spread of information at a two different scales, at the level of individual tweets and at a higher level that includes all topically related tweets. Finally, we include the temporal dimension in the higher level graph model of the Tweet data set.

3. Classifier Design: We design a classifier for Twitter data sets of different countries based on features of their information cascades. This classifier is able to distinguish Twitter traffic pertaining to countries experiencing political upheaval and social unrest from those that are not.

[1]This data will be made publicly available, following its acceptance for publication.

Chapter 2

Related Work

2.1 Related Work

Broadly speaking the previous work related to our work can be categorized into two groups.

2.1.1 Information Cascades

The first category of prior work focuses on different methodologies to define information cascades in a social network. The second category of prior work aims to identify unique patterns across different topics in online social networks.

In prior literature, researchers have defined information cascades in different ways. This is evident by use of cascade in different manner in recent literature which include, [21], [26], [6], [12] and [30]. The general consensus is to model information cascades as graphs where nodes represent users and edges represent interaction among users. There is no single agreed upon definition of what constitutes an information cascade.
Several of those definitions were developed in the context of data obtained from Twitter. For instance, Bakshy *et al.* defined information cascades as all the data related to an event, news, or URLs [3]. They used this particular definition of information cascades to track the dissemination of a URL over users' Twitter follower graph. In another work Kwak *et al.* performed a measurement study of tweets related to a specific topic [20]. They defined an

information cascade as the set of tweets and retweets containing a specified set of terms, *e.g.* Air France Flight. In [25], Sadikov *et al.* discussed various methods to construct information cascades in online social networks. An interested reader is referred to [25] for a more detailed discussion.

2.1.2 Pattern Characterization

Detecting unique patterns shown by different topics in online social networks has gained a lot of interest in previous years. Some of the recent work targeting the detection of unique patterns in social network graphs includes [19], [17], [31], [5], [1] and [2].
It has been widely believed that user interactions about different topics exhibit different propagation characteristics as they propagate over a social network. One set of empirical results to validate this hypothesis has been provided in [23] by Romero, Meeder and Kleinberg. They showed that information propagation characteristics vary for different types of topics. However, they did not investigate the difference in propagation characteristics for information related to similar topics. In another work, Gong *et al.* also observed the differences in information dissemination characteristics for different topics [13]. They analyzed the properties for different temporal snapshots of the social network graph instead of analyzing it in aggregate. In another related work, Iliofotou *et al.* investigated dynamic graphs for traffic classification in computer networks [18]. The use of dynamic graphs is not explored in details for online social network analysis. In this paper, we will utilize a similar dynamic graph based approach to detect countries experiencing political unrest using Twitter traffic.

Some recent studies have focused on finding patterns on tweets related to political campaigns. Among these kind of works Avishey Lavine et.al [22] have used tweets generated by candidates during 2010 US midterm elections and have pointed out difference of graph structure and content generated by candidates with different party affiliations. Avishey Lavine *et al.* in [22] also used the structural and content properties to predict the successful candidates. In other works by Sarita Yardi and Danah Boyd in [32] and M.D.Conover *et al.* in [8, 7] have used Twitter platform to discriminate between users based on different party affiliations, they have utilized the properties of content within tweets as well as graph properties of user interaction behavior to detect the political polarisation.

We have now seen that previous work have either focused on detect-

ing differences between traffic of different types of topics or within same topics people have tried to detect the different type of users. But no previous work has worked on detecting *political stability* across different countries and detecting political stability is the main focus of this paper. Most of the previously used methods are not scalable, due to their reliance on tweet content for detecting patterns and content based properties will not be helpful when input data is comprising of multitude of languages. To take care of this shortcoming our methods do not rely on tweet text for the classification purposes.

Chapter 3

Data Set

Twitter is very open to data collection, to the extent that it encourages it. Data sets like those from Twitter are rare in the sense that they are not anonymized or redacted. It allows access to almost all data associated with a tweet, including message contents. The Twitter data used in this study was collected using the Python [16] *tweetstream* package [14] which provides access to Twitter's Streaming API. Tweets from the streaming API were filtered by a list of country names. This way, the collected tweets either contained the name of a country in the tweet body or in their associated metadata. The list of countries we targeted for data collection comprises of countries in the ME/NA regions which were experiencing instability, political upheaval and popular uprising during the time now referred to as the 'Arab Spring'. Our data set comprises of two sets of countries. One set consists of ME/NA countries which were going through the phase of social unrest throughout the spring of 2011 and beyond. The second set was collected for countries which were most definitely not in a state of political unrest. Data collection for ME/NA countries commenced in March and, with some breaks in between, continued through the end of June, 2011. In this work, we used data from the one week period from April 11-17, 2011. The reasons we did not use the entire data set and chose this particular time period were two-fold:

1. **Data Set Size:** When the duration of the data set is increased to more than about 7 days, the computing requirements for processing it exceed the capabilities of the high-performance computing facilities available to the authors.

2. **Peak Activity:** Over the time periods for which we collected data, social unrest in the ME/NA was (arguably) at a peak.

We carried out a second phase of data collection for countries not experiencing any political instability. For this purpose a set of eight countries was selected and data collection was carried out for a 7 day duration. The second phase commenced on July 4 and ended on July 11, 2011. To verify the generality of our classification method we performed a third phase of data collection. In this third phase we collected Twitter traffic on trending topics that were unrelated to our previous data sets. These additional data sets can be classified into two groups; a) News & politics and b) movies. In the news and politics category we collected tweets mentioning three candidates in the republican primary election for the 2012 U.S. presidential election; Herman Cain, Mitt Romney and Rick Perry. We also collected tweets on the issue of 'European debt' and the 'Euro zone.' In the category of movies we collected tweets for three upcoming titles that were apparently trying to create buzz through a Twitter campaign. These included 'Paranormal Activity' (3), 'Real Steel' and 'The Three Musketeers.'

Table 3.1 provides a summary of the list of keywords for which Twitter data was collected, the number of tweets and unique users each contains, the average number of tweets per user, the percentage of tweets that are retweets or replies and the percentage of tweets that contain #-tags, URLs and mentions.

Table 3.1: Basic statistics of Twitter data sets.

Class	Country	Unique Users	Tweet Count	Tweets per User	Retweeted Tweets (%)	Reply Tweets(%)	Tweets w/ #-tag(s) (%)	Tweets w/ URL(s) (%)	Tweets w/ Mention(s) (%)
Stable (7-10 Jul, 2011)	Australia	100353	170378	1.70	19.93	18.50	25.23	43.97	49.58
	Austria	13450	21164	1.57	20.86	17.68	28.63	38.59	51.55
	New Zealand	17736	28015	1.58	14.53	13.82	19.29	55.55	37.53
	China	170346	269798	1.58	17.89	12.97	17.53	48.34	43.19
	Poland	10973	19422	1.77	14.67	28.52	25.77	37.84	50.61
	Iceland	7449	10075	1.35	17.14	17.06	23.04	50.84	45.76
	Turkey	61657	83338	1.35	13.21	21.56	21.42	24.85	47.29
	Norway	9578	14695	1.53	13.09	29.81	21.75	39.99	53.36
	Japan	146922	246450	1.68	22.34	8.58	41.84	52.25	44.90
Unstable (11-17 Apr, 2011)	Bahrain	30958	324878	10.49	60.25	6.35	78.11	37.39	72.48
	Egypt, Cairo	68873	379112	5.50	39.08	6.51	74.57	46.02	53.12
	Syria	29428	203923	6.93	52.97	7.29	90.02	46.70	68.01
	Libya, Tripoli	63714	301915	4.74	38.20	6.29	57.39	54.75	53.44
	Oman	114316	34674	2.42	41.23	10.55	58.87	39.50	64.48
	Yemen	16924	58489	3.46	34.15	7.76	67.99	53.30	49.21
	Saudi Arabia	79937	182487	2.28	16.92	22.87	38.11	14.56	50.77
	Iran, Tehran	34035	147545	4.34	26.23	6.74	70.24	66.67	43.19
News & Politics (27 Oct -4 Nov, 2011)	Herman Cain	66385	138571	2.09	35.31	3.62	17.51	48.46	54.88
	European Debt	4137	5874	1.42	13.89	0.58	16.63	85.50	23.19
	Euro Zone	19752	33423	1.69	26.30	3.00	31.60	61.82	39.69
	Rick Perry	22801	39520	1.73	31.65	4.59	20.23	56.71	52.56
	Romney	29513	64205	2.18	30.06	8.77	23.66	52.10	54.30
Movies (27 Oct-4 Nov, 2011)	Paranormal	151483	186708	1.23	14.38	9.76	15.08	6.77	40.86
	Real Steel	23877	32272	1.35	4.51	13.86	8.43	16.07	51.27
	The Three Musketeers	5115	6552	1.28	10.09	8.26	14.84	38.49	46.08

Chapter 4

Technical Approach

In this chapter, we provide the details of our proposed methods to detect political unrest using Twitter data.

As mentioned in Section 3, Twitter allows users to post messages (or tweets) each containing up to 140 characters. From now-onwards, we refer to the user who originally posted a tweet as *initiator*. The tweets are immediately available in "timelines" of the followers of initiators. Furthermore, the subset of these tweets which are made public also appear in a public timeline which is accessible to all other users in the social network. Twitter users can interact with any visible tweet in two ways: *retweet* and *reply*. In case of retweet, an exact copy the original tweet is posted on their profile and it is also visible to their followers' timeline. In case of reply, a custom tweet is created containing the tag of the initiator. In addition to the aforementioned two ways, users can also *mention* (*i.e.* tag) other users in their tweets. In a given tweet's mention field, multiple users can be mentioned. Note that replies are equivalent to mentions containing one tag. In addition to up to 140 character text, tweets also contains a timestamp and user's profile information. They may also optionally contain hashtags and URLs. If a tweet is retweet, reply, or mention then it will also contain information of all referred users.

The interactions of users with tweets is equivalent to information flow over the social network. Each set of these interactions can be conceptually represented as an *information cascade*. Note that an information cascade can be represented using graph data structure, where nodes are users and edges represent timestamped interactions among users. However, there are different ways at which these information cascades can be defined. For instance, we

may only consider a tweet and its retweets as one information cascade. At the other extreme, we can also consider all tweets, retweets, replies, and mentions containing a specific hashtag or text keyword as an information cascade.

Recall that our goal is to classify countries based on their Twitter traffic. We aim to identify and explore differences among information cascades observed in their traffic. In this paper, we present four different methods using which we can analyze differences among information cascades pertaining to different countries. These methods differ in terms of how we define an information cascade and what properties we compute from them. We describe each of them separately in the following text.

4.1 Individual Tweet Analysis (M1)

In this analysis method (M1), we treat each set of tweets, retweets, and replies as an information cascade. Consider a single original tweet m by a Twitter user v. Let $\mathbf{R}(m)$ be the set of Twitter users that retweeted tweet m. Let $\mathbf{P}(m)$ be the set of Twitter users that replied to tweet m. Now the information cascade is the graph comprising of the set of nodes $v \cup \mathbf{R}(m) \cup \mathbf{P}(m)$ and the set of directed edges among them based on interaction information. Note that we only include retweet and reply interactions as edges in these information cascades.

Using the above-mentioned methodology, we can obtain information cascade graphs for all countries in our data set. We now analyze each of these information cascade graphs using their basic properties. We analyze two properties in this method: (1) edge count and (2) diffusion lifetime. We define both of them below.

4.1.1 Edge Count

Let $G = (V, E)$ denote the graph of an information cascade then edge count is simply the number of elements in the edge set ($|E|$).

4.1.2 Diffusion Lifetime

The diffusion lifetime (DL) is related to the time duration between the appearance of the original tweet and the appearance of the last retweet in an

information cascade. Given the information cascade graph G with node set V and edge set E, and let t_{ij} denote the timestamp corresponding to the edge e_{ij}, then the tweet diffusion lifetime is defined as:

$$DL = P_{90}\left(\bigcup_{v_i,v_j} t_{ij}\right) - \min\left(\bigcup_{v_i,v_j} t_{ij}\right), \tag{4.1}$$

where, $P_{90}(\cdot)$ is an operators that returns the 90^{th} percentile value and $min(\cdot)$ is the minimum operator.

The aforementioned properties can be computed for each information cascade graphs pertaining to a country. We need to define a manageable number of "features" based on these properties that can further be used for classification. Towards this end, we first plot probability density functions (PDFs) of these features and observe that these distributions are highlight skewed (see Figure 6.1). In particular, we note that they are heavy-tailed and roughly follow a straight line when both axes are converted to the logarithmic scale. Given that these properties follow heavy-tailed distribution, we can quantify their characteristics using a single parameter called scaling exponent. Thus, we characterize all information cascades pertaining to a country in terms of the scaling exponents of their feature distributions.

4.2 Multiple Tweet Static Analysis (M2)

In this analysis method (M2), we treat all tweets, retweets, replies, and mentions specific to a topic or set of keywords in a given time period as an information cascade. More specifically, let $\mathbf{m}$ denote the set of all original tweets by all users $\mathbf{v}$ during the time period T related to some topic. Also let $\mathbf{R}(\mathbf{m})$ and $\mathbf{P}(\mathbf{m})$ be the set of Twitter users that retweeted or replied to the tweets $\mathbf{m}$. Furthermore, let $\mathbf{M}(\mathbf{m})$ be the set of Twitter users that mentioned users any of the users in $v \cup \mathbf{R}(\mathbf{m}) \cup \mathbf{P}(\mathbf{m})$. The information cascade graph using multiple tweet static analysis comprises of the set of nodes $v \cup \mathbf{R}(\mathbf{m}) \cup \mathbf{P}(\mathbf{m}) \cup \mathbf{M}(\mathbf{m})$ and the set of directed edges among them based on interaction information. In contrast to individual tweet analysis method, we additionally include mention interactions as edges in these information cascades. For information cascades constructed in this method, we select the time period $T = 24$ hours. We believe that this time period is long enough to provide us a significant majority of all tweets related to any topic. It is

shown in prior literature that about 75% of retweets appear within the first 24 hours of the appearance of the original tweet in Twitter [20].

Using the above-mentioned methodology, we can obtain information cascade graphs for all countries in our data set. To analyze these information cascade graphs, we need to analyze their basic structural properties that are defined for individual nodes. The set of basic properties we analyze for information cascades in this method are: (1) node degree, (2) node out-degree, (3) node in-degree, (4) closeness centrality, (5) clique count, and (6) triangle count. Below we define each of them separately. For introduction to graphs and their properties, interested reader is referred to [27].

4.2.1 Node Degree

The degree of a node is defined as the number of edges incident on it. The degree (δ_i) of a node v_i is defined as:

$$\delta_i = \left| \bigcup_{\forall j=i \vee k=i} e_{jk} \right|, \tag{4.2}$$

where e_{jk} denotes an edge between nodes j and k.

4.2.2 Node In-Degree

Similarly, the in-degree of a node is defined as the number of incoming edges incident on it. The in-degree ($\delta_{\downarrow_i}$) of a node v_i is defined as:

$$\delta_{\downarrow_i} = \left| \bigcup_{\forall k=i} e_{jk} \right| \tag{4.3}$$

4.2.3 Node Out-Degree

Likewise, the out-degree of a node is defined as the number of outgoing edges incident on it. The out-degree ($\delta_{\uparrow_i}$) of a node v_i is defined as:

$$\delta_{\uparrow_i} = \left| \bigcup_{\forall j=i} e_{jk} \right| \tag{4.4}$$

4.2.4 Closeness Centrality

The closeness centrality of a node is defined as the average length of shortest paths to all nodes reachable from it. Let l_{ij} denote the shortest path length from node v_i to node v_j. Closeness centrality was defined for the first time by Sabidussi in [24]. The closeness centrality (c_i) of a node v_i is defined as:

$$c_i = \frac{\sum_{j=1}^{N} l_{ij}}{|V|} \tag{4.5}$$

4.2.5 Clique Count

A clique is defined as the subset of nodes in a graph that form a complete graph. The clique count of a node is simply the number of cliques it is a part of.

4.2.6 Triangle Count

A triangle is defined as the subset of any three nodes in a graph that are completely connected. The triangles count of a node is defined as the number of triangles it is part of. Let Γ_i denote the set of nodes that a node v_i is connected to then the triangle count (Δ_i) is defined as:

$$\Delta_i = \left| \bigcup_{v_j, v_k \in \Gamma_i} e_{jk} \right| \tag{4.6}$$

In contrast to information cascades constructed for individual tweet analysis method (see Section 4.1), these information cascade graphs are much larger in terms of number of nodes and edges. Furthermore, unlike the first method (M1), each information cascade graph here may not be completely connected as a single component. Therefore, we can split one information cascade graph into its components before computing its structural properties. Note that the structural properties can be computed for each node of all components of information cascade graphs pertaining to a country. To map a large number of these values for each component to a more manageable number of features, we again utilize the fact that PDFs of these features have heavy-tailed distribution and use the scaling exponent to quantify their characteristics (see Figure 6.2).

4.3 Multiple Tweet Dynamic Analysis (M3)

In this analysis method (M3), we construct information cascades in the same way as multiple tweet analysis in Section 4.2. However, we now study information cascade graphs in a pairwise manner. More specifically, we compute properties across consecutive 24-hour time sliced information cascade graphs. Note that these properties are different than those computed for previously mentioned methods.

We introduce some notation to define properties that are computed across consecutive time sliced information cascade graphs. Let $G[t]$ denote a sequence of graphs, where $1 \leq t \leq \mathtt{T}$, where $\mathtt{T}$ denotes the number of time slices. Note that the duration of each time slice is 24 hours. The set of properties we analyze in this method are: (1) relative inclusion, (2) Jaccard index, and (3) graph edit distance. We define them in the following text.

4.3.1 Relative Inclusion

Relative inclusion (RI) is can be defined based either on nodes or on edges. For two time-neighboring information cascade graphs $G[t] = (V_t, E_t)$ and $G[t+1] = (V_{t+1}, E_{t+1})$, relative inclusion is defined as the ratio of the number of nodes present in the union of graphs $G[t]$ and $G[t+1]$ to the number of nodes in the first graph $G[t]$.

$$RI = \frac{|V_t \cap V_{t+1}|}{|V_t|} \tag{4.7}$$

Similarly, relative inclusion can be defined for number of edges as:

$$RI = \frac{|E_t \cap E_{t+1}|}{|E_t|} \tag{4.8}$$

4.3.2 Jaccard Index

The Jaccard index, also known as Jaccard similarity index, is primarily used to quantify the similarity of two sets. The Jaccard index is a normalized

measure of the overlap in the number of vertices or edges for two time-neighboring information cascade graphs. For graphs $G[t]$ and $G[t+1]$ defined above the Jaccard index (JI) for nodes is defined as:

$$JI_V = \frac{|V_t \cap V_{t+1}|}{|V_t \cup V_{t+1}|} \tag{4.9}$$

Similarly, the Jaccard Index is defined for edges as:

$$JI_E = \frac{|E_t \cap E_{t+1}|}{|E_t \cup E_{t+1}|} \tag{4.10}$$

4.3.3 Graph Edit Distance

The graph edit distance is defined as the number of edit operations that have to be performed on one graph to make it same as another graph. Graph edit operations include addition and removal of nodes or edges. For two graphs $G[t]$ and $G[t+1]$, the graph edit distance is defined as:

$$\begin{aligned} D(G[t], G[t+1]) = &|V_{t+1} - V_t| + |V_t - V_{t+1}| \\ &+ |E_{t+1} - E_t| + |E_t - E_{t+1}| \end{aligned} \tag{4.11}$$

Note that properties computed in this method are computed between time-neighboring information cascade graphs. If we have a total of `T` slices for a country then we get only $\mathtt{T} - 1$ data points for it. Due to sparsity of points, we cannot use the feature computation method used in previous two methods (M1 and M2). In this method, we directly use the properties as features for classification.

4.4 Tweet Content Analysis (M4)

This method is used for baseline comparison, where we only use aggregate properties of tweets for each country as features. The six properties that we use in this method are: (1) average tweets per user, (2) re-tweeted tweets percentage, (3) reply tweets percentage, (4) hashtag tweets percentage, (5)

URL tweets percentage, and (6) mention tweets percentage. Note that properties computed in this method are computed for each 24 hour slice of all countries. We use these properties directly as features for classification.

4.5 Classification

We now provide details of the classification methodology used to classify features from the aforementioned four methods for each country into two classes: stable and unstable. We first do feature selection to remove redundant features and then use standard machine learning classification algorithms for classification.

In feature selection, our goal is to rank features based on their differentiation power across classes. Towards this end, we use the following 3 separate entropy based metrics for rank features [10].

For our two-class classification problem, we tried multiple machine learning classification algorithms In this paper, we only report the results of three of them which provided better classification accuracy. These three classifiers are: Naïve Bayes, SVM and Bayesian logistic regression. We report the accuracy of these classification algorithms in terms of the following four metrics. The true positives (TP) rate is the ratio of the number of instances that it correctly labels as belonging to the positive class to the total number of positive class instances. The false positive (FP) rate is the ratio of the number of instances that it incorrectly labels as belonging to the positive class to the total number of positive class instances. The precision of a classifier is the ratio of the number of instances that it correctly labels as belonging to the positive class to the total number of instances labeled as belonging to the positive class. The accuracy of a classifier is the ratio of the number of instances that it correctly labels to the total number of instances. To report classification results in terms of the aforementioned metrics, we use the standard N-fold cross validation procedure. We have used $N = 10$ for all classification experiments in this study.

Chapter 5

Classification

In this chapter we will explain the machine learning algorithms used for classifying data for stable and unstable countries. We have used supervised learning algorithms for classification in this thesis. For further details on the used classification algorithms interested reader is referred to [4, 29, 9]

5.1 Overview

Classification problem can be sub-divided into smaller sub-problems, in this overview we will discuss these sub-classes of a classification problem breifly:

1. Data Aquisition: First step for solving a classification problem is the collection of data, we collect data in raw form and perform pre-processing operations to get a form suitable for applying classification algorithms. For example, in the context of this work we had used API provided the Twitter streaming application for collection of Tweets. The tweets collected, provided us with a lot of data like tweet text, tweet user, urls, mentions, etc. But in this form this data is of no use since we can not use it for solving a classification problem at hand.

2. Feature Extraction: After data is being collected our aim is to extract features for to be used for classification purposes. Consider the example of unrest detection problem we have solved in this thesis, to extract features from the collected Twitter data we extract graphs for tweet propagation and user interaction behavior using the data present within collected tweets. The properties of these graphs are then used to define

the features of classification problem at hand.

3. Classifier Design: After successful extraction of features from the data set we are finally in a position to solve the problem as a classification problem. For the purpose of classification we use algorithms like Support Vector machine etc, brief overview of classification algorithms used in this work is provided in next sections.

5.2 Classifiers

Based on 10-fold cross validation we have selected naive Bayesian, logistic regression and support vector machines for classifying data. In the following explanations $x_j^{(i)}$ denote the i^{th} training sample of the j^{th} feature and $y^{(i)}$ is used to represent the label of the i^{th} training sample.The details about working of these algorithms is provided in following sub-sections:

5.2.1 Logistic Regression

For input features x_j, the aim of linear regression analysis is to find a line that fits the values of x_j as function of y. Objective is to find the parameter θ such that we find h_θ which as close to our original function y as possible. For linear regression h_θ is defined as:

$$h_\theta(x) = \theta_0 + \sum_{j=1}^{n} \theta_j x_j \tag{5.1}$$

We have defined the basic idea behind regression analysis in 5.1, now lets turn our attention towards logistic regression. Logistic regression is a classification method which mean instead of modeling relation between x and y our main aim is to classify feature value x to correct label y. In case of binary classification, to fulfill the aim of classifying data we apply sigmoid function or logistic function to our input sequence x. Logistic function is defined as:

$$h_\theta(x) = g\left(\theta^T x\right) = \frac{1}{1 + e^{-\theta^T x}} \tag{5.2}$$

The value of this function will be approaching -1 in case of negative class and +1 in case of positive class.

5.2.2 Naïve Bayesian

Naïve Bayesian algorithm belongs to the class of generative algorithms e.g. for binary classification problem we model probability distributions for both the classes based on training data and new data point for classification is compared with both the models to decide upon the class. Naïve Baye's algorithm works under the assumption of conditional independence also called Naïve Bayes assumption.. It is assumed that features x_j are conditionally independent given y. Probability distribution for feature x given y is given by:

$$p\left(x_1, x_2, ..., x_n|y\right) = p\left(x_1|y\right) p\left(x_2|y, x_1\right) ... p\left(x_n|y, x_1, ..., x_{n-1}\right) \tag{5.3}$$

Using Naïve Bayes assumption:

$$p\left(x_1, x_2, ..., x_n|y\right) = \prod_{j=1}^{n} p\left(x_j|y\right) \tag{5.4}$$

Our model parameters are $\phi_{j|y=1}$, $\phi_{j|y=0}$ and ϕ_y, and these are defined as:

$$\phi_{j|y=1} = p\left(x_j = 1|y = 1\right) \tag{5.5}$$

$$\phi_{j|y=0} = p\left(x_j = 1|y = 0\right) \tag{5.6}$$

$$\phi_y = p\left(y = 1\right) \tag{5.7}$$

Values for these parameters can be estimated using maximum likelihood estimator:

$$\mathcal{L}\left(\phi_y, \phi_{j|y=0}, \phi_{j|y=1}\right) = \prod_{j=1}^{m} p\left(x^{(i)}, y^{(i)}\right) \tag{5.8}$$

Once these values are known we can use Baye's law to find out the aposterior probability value:

$$p\left(y = 1|x\right) = \frac{p\left(x|y = 1\right) p\left(y = 1\right)}{p\left(x\right)} \tag{5.9}$$

5.2.3 Support Vector Machine (SVM)

To understand the working of SNM we will start by understanding the concept of *margin*, the main goal while designing the SVM classifier is to maximize the margin and hence improving the classifier efficiency.

1. Margins: Consider a binary classification problem and let:

$$p = \left(\omega^T x^{(i)} + b\right) \tag{5.10}$$

be the linear hyperplane separating the two classes. Here, ω^T is the vector representing the slope of hyperplane and b is the intercept value. **Functional margin** of i^{th} sample is defined as:

$$\hat{\gamma} = y^{(i)} p \tag{5.11}$$

For complete data set the functional margin γ is defined as:

$$\gamma = \min \hat{\gamma}^{(i)} \tag{5.12}$$

The problem with functional margin is that scaling ω and b will give us scaled value of margin, while in reality there is no effect on margin. In order to take care of this problem we normalize value of functional margin by $\|\omega\|$, by normalizing functional margin we get **geometric margin**:

$$p = \left(\frac{\omega^T x^{(i)}}{\|\omega\|} + \frac{b}{\|\omega\|}\right) \tag{5.13}$$

2. Margin maximization: After defining margin, we turn to main goal of classification problem i.e. maximizing the margin. For optimization right now are assuming linearly separable data. The optimization problem looks like:

$$\begin{aligned} &\max_{\gamma,\omega,b}\ \gamma \\ \text{s.t. } &y^{(i)}\left(\omega^T x^{(i)} + b\right) \geq \gamma \\ &\|\omega\| = 1 \end{aligned} \tag{5.14}$$

Optimization equation in this form is not desireable since it is not a convex optimization problem due to the second constraint, which is a non-convex constraint. After performing some mathematical operations we transform the problem into a convex optimization problem:

$$\begin{aligned} &\min\ \frac{1}{2}\|\omega\|^2 \\ \text{s.t. } &y^{(i)}\left(\omega^T x^{(i)} + b\right) \geq \gamma \end{aligned} \tag{5.15}$$

We will now represent the problem in dual form and using Karush-Kuhn-Tucker(KKT) conditions we will see that $p^* = d^*$, where p^* and

d^* are the solutions for primal and dual problems respectively.

$$\begin{aligned} \max_\alpha W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha^{(i)} \alpha^{(j)} \langle x^{(i)}, x^{(j)} \rangle \\ \text{s.t. } \alpha_i \geq 0,\ i = 1, 2, ..., m \\ \sum_{i=1}^{m} \alpha_i y^{(i)} = 0 \end{aligned} \tag{5.16}$$

Another important point obtained from dual form is that third KKT-condition is zero for all the values of data set except for the **support vectors**, hence we do not need to solve the problem for values other than support vectors, resulting in decrease of complexity with increasing data set size. Another advantage of dual representation is that we can represent the optimization problem in kernel form.

3. Kernels: Untill now we have considered that data is linearly separable. But in reality this assumption might not hold. Kernels can be used to aleviate this problem of linear separation to some extent, by mapping input attributes x of data set to a higher dimensional feature space $\phi(x)$.

5.3 Feature Ranking Methods

We have use three entropy based feature ranking methods to rank the features in the order of their usefulness. These methods are explained below. For further detail on the topic interested readers are recommended to read [10].

1. **Information Gain (IG)** is defined in terms of another information theoretic measure called mutual information. The mutual information of any two random variables (or features) X_i and X_j is defined as:

$$I(X_i; X_j) = H(X_i) - H(X_i|X_j), \tag{5.17}$$

where, $H(X_i)$ is entropy of feature X_i, and $H(X_i|X_j)$ is the conditional entropy of X_i given X_j. The information gain for feature X_i with respect to class variable $\mathbf{Y}$ is defined as:

$$IG(X_i) = \sum_{\forall j} I(X_i; Y_j) \tag{5.18}$$

2. **Gain Ratio** (GR) is a normalized form of mutual information. It is defined as the ratio of mutual information between feature and class variable to the entropy of feature:

$$GR(X_i) = \frac{I(X_i; \mathbf{Y})}{H(X_i)} \tag{5.19}$$

3. **Symmetrical Uncertainty** (SU) is the sum of normalized measures of pairwise mutual information of feature and class variable. Symmetrical uncertainty of a feature X_i is defined mathematically as:

$$SU(X_i) = \sum_{\forall j} 2 \left[\frac{I(X_i, Y_j)}{H(X_i) + H(Y_j)} \right] \tag{5.20}$$

5.4 N-fold cross validation

We use N-fold cross validation to evaluate the performance of the classifiers. Cross validation involves spliting of example data into two data sets, we use one partition of data set for training purposes (e,g. 70% of data) and the second partition (e,g. remaining 30% data set) as a training set. To further improve the performance of cross validation framework we use N-fold cross validation. In this case we split data into N partitions and use on of the partition as a test sequence and rest of the partitions as training data set. We use all the combinations for making one of the partition as test sequence while using remaining partitions as training sequence and we average out the performance for all the iterations.

Chapter 6

Evaluation

In this chapter, we evaluate the effectiveness of our proposed approaches to detect political unrest in Twitter data. We provide evaluation results of all approaches separately in the following subsections.

6.1 Individual Tweet Analysis (M1)

Recall that we used utilized two individual tweet properties: edge count and diffusion lifetime, to extract features. We then used the slopes of heavy tailed distributions as eventual features for classification. We plot the PDFs of both features in Figure 6.1. The distributions indeed seem to follow a heavy tailed distribution as they follow a straight-line on logarithmic x and y scales. The goodness of fit values R^2 are provided, which indicate a reasonable fit for all categories. Across the two classes, *i.e.* stable and unstable, we note that the values of slopes do not show a clear trend. For instance, the slope of the fitted line for stable distribution is larger than that of unstable distribution for edge count feature in Figure 6.1(a). This trend is reversed for the diffusion lifetime feature, where the slope of the fitted line for stable distribution is smaller than that of unstable distribution. We also observe that the politics category also shows some randomness. Note that for eventual classification we merge the politics category into the stable class to make sure that our classification method is not just detecting unstable countries due to "trending events". We further use these features with machine learning algorithms for classification. The classification results of this method (M1)provided in Table 6.3 also concur our observations from Figure 6.1. The best results are obtained for naïve

Bayes, which has TP rate of 0.500 and a FP rate of 0.00. This shows that analyzing individual tweets does not reveal any distinguishing characteristics across information cascades of politically stable and unstable countries.

6.2 Multiple Tweet Static Analysis (M2)

Recall that we analyzed a set of six node-level properties to characterize information cascades in multiple tweet static analysis. Similar to individual tweet analysis, we decided to use heavy-tailed distribution slopes as features for classification. Figure 6.2 shows the plot of distributions of the six features for stable countries, unstable countries, and politics category. We observe that most feature distributions follow the heavy-tailed distribution. The only exception is closeness centrality which shows a unique bimodal structure. Therefore, R^2 values for closeness centrality are significantly lower than other features, as shown in figure 6.3. From Figure 6.2, we also note that slope values for stable countries show difference compared to those for unstable countries.

To further analyze these trends, we plot the conditional distribution plots of all features in Figure 6.4. We observe different trends across different features. For instance, the slope values are larger for unstable countries than those for stable countries in case of node degree and node in-degree features. On the other hand, the slope values are smaller for unstable countries than those for stable countries in case of closeness centrality and clique count features. We do not observe a clear separation for node out-degree and triangle count features.

Table 6.1: Feature ranking for multiple tweet static analysis method (M2).

Property	Gain Ratio	Information Gain	Symmetric Uncertainty
Degree	2	1	1
In-degree	1	2	2
Cliques	3	4	3
Closeness	6	3	4
Triangle	5	5	5
Outdegree	4	6	6

To quantitatively study the quality of different features for classification, we use the information-theoretic features explained in chapter 5. We use the trio of information gain IG, gain ratio GR, and symmetric uncertainty SU to rank features according to their classification power. Table 6.1 shows that ranking of all features with respect to the three ranking measures. We observe that degree and in-degree are ranked as the beast features by all of our ranking metrics. As we observed in Figure 6.4, the ranking results also show that node out-degree and triangle features have the lowest differentiation power.

Now now jointly use all of these features with machine learning algorithms for classifying politically stable and unstable countries. Table 6.3 shows the classification results for all algorithms using this method (M2). We first note that these results are significantly improved as compared the results of individual tweet analysis method (M1). This highlights that analyzing information cascades of multiple tweets indeed provides additional useful information. Across classifiers, we again observe that naïve Bayes provides the best performance across all of our accuracy metrics.

6.3 Multiple Tweet Dynamic Analysis (M3)

We now analyze dynamic properties of multiple tweet method for classifying politically stable and unstable. Recall that in dynamic analysis of multiple tweets we computed features across consecutive time sliced information cascade graphs. These features included relative inclusion for nodes and edges, Jaccard index for nodes and edges, and graph edit distance.

To study the features across two classes, we first plot their conditional distributions in Figure 6.5. We note that relative inclusion and Jaccard index based features provide clear difference across stable and unstable countries. We note that the values of these four features for unstable countries are consistently larger than those for stable countries and politics (see Figure 6.5(a)–(d)). These feature results show that same nodes and edges appear in consecutive information cascade graphs for unstable countries. On the other hand, the probability of occurrence of a node or an edge across two consecutive days is lower for stable countries. These observations are in accordance with findings in the prior work. For example, Gong *et al.* observed that if consecutive information cascades contain similar content then the structure of information cascade graphs also remain similar [13]. In other words, the probability of occurrence of same nodes and edges across consecutive days

Table 6.2: Feature ranking for multiple tweet dynamic analysis method (M3).

Property	Gain Ratio	Information Gain	Symmetric Uncertainty
Jaccard Index – nodes	1	1	1
Relative Inclusion – nodes	2	2	2
Jaccard Index – edges	4	3	3
Relative Inclusion – edges	3	4	4
Graph Edit Distance	5	5	5

is significantly higher is contents of information cascades do not change. Likewise, Kwak *et al.* studied the difference between the size of active user population for different trending topics. Their experimental results showed that cumulative unique user count became almost constant after few days for tweets related to Iran. In contrast, the cumulative unique user count continued to increase for the trending topic 'Apple'. In our proposed method, relative inclusion and Jaccard index based features are essentially capturing the similar information. The only feature that does not provide clear distinction among stable and unstable countries is graph edit distance.

We now rank of these features using the information theoretic metrics mentioned in Section 4.5. Table 6.2 shows the rankings of the five features using the aforementioned three metrics. As expected, we observe that relative inclusion and Jaccard index based features consistently rank the highest with respect to all three metrics. We also observe that node based features have higher ranks than edge based features in Table 6.2. This highlights that presence of same users across consecutive days has more relevant information than the presence of same interactions or edges.

We now use these five features with standard machine learning classifier for automated detection of politically unstable countries. Table 6.3 provides the accuracy results of all classifiers for this method (M3). We note that both naïve Bayes and SVM classifiers provide close to perfect accuracy. These classification results are significantly better than those of individual tweets analysis and multiple tweet static analysis methods. This shows that analyzing information cascades of multiple tweets using dynamic features provides about 99% accuracy, with almost perfect TP and FP rates.

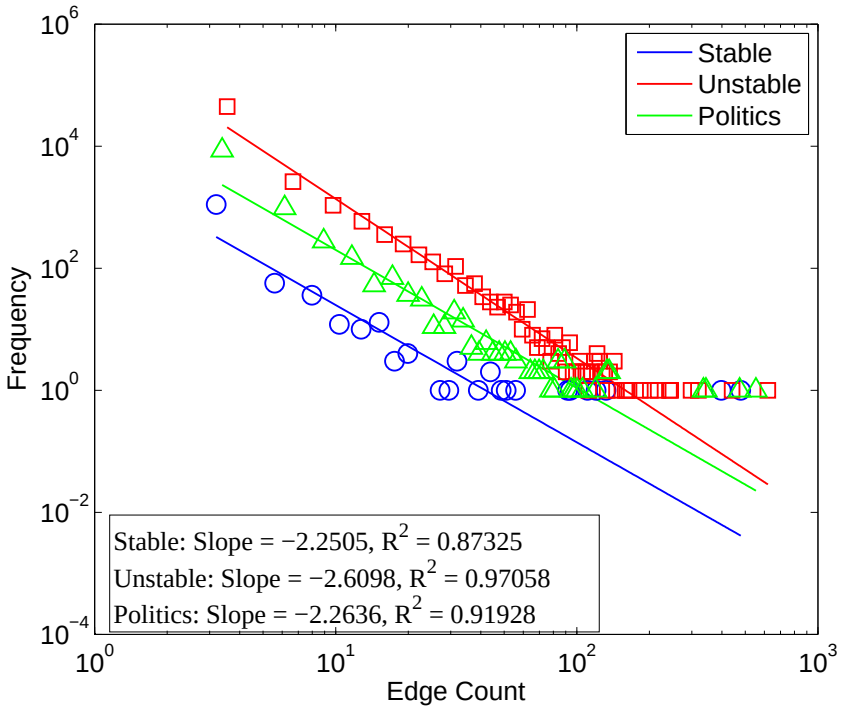
6.4 Tweet Content Analysis (M4)

For baseline comparison, we now evaluate the classification performance of tweet content analysis method. Recall that we used content based features (included in Table 3.1) used in our study are average tweets per user, retweeted tweets percentage, reply tweets percentage, hashtag tweets percentage, URL tweets percentage, and mention tweets percentage. It is undesirable to use these features because they may require manual labeling of some features and therefore cannot be fully automated. This problem is particularly important if the content is available in multiple languages, which is a common case for Twitter allowing users to upload content in more than 17 languages.

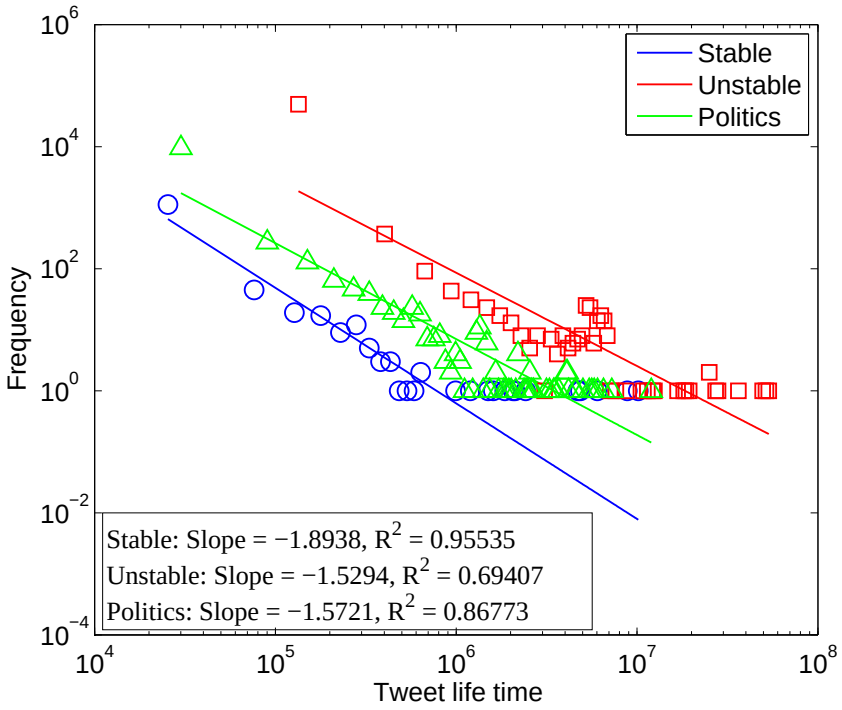
Table 6.3 tabulates the classification results of the naïve Bayes, SVM, and Bayesian logistic regression algorithms. We observe that naïve Bayes consistently performs other classification algorithms in terms of accuracy, where we obtain approximately 90% TP rate and 5% FP rate. Although its classification performance is better than the first two methods (individual tweet analysis and multiple tweet static analysis), it is still not as good as that of multiple tweet dynamic analysis method.

Table 6.3: Classification results of all methods used in our study. M1 is the individual tweet analysis method, M2 is the multiple tweet static analysis method, M3 is the multiple tweet dynamic analysis method, and M4 is the tweet content analysis method.

Metric	**Method**	**Naïve Bayes**	**Support Vector Machine**	**Bayesian Logistic Regression**
TP rate	M1	0.500	0.000	0.000
	M2	0.893	0.875	0.982
	M3	**1.000**	**1.000**	1.000
	M4	0.875	0.875	0.750
FP rate	M1	0.000	0.000	0.000
	M2	0.282	0.205	0.410
	M3	**0.030**	**0.000**	1.000
	M4	0.031	0.052	0.010
Precision	M1	1.000	0.000	0.000
	M2	0.820	0.860	0.775
	M3	**0.980**	**1.000**	0.530
	M4	0.942	0.907	0.977
Accuracy	M1	0.750	0.500	0.500
	M2	0.810	0.835	0.786
	M3	**0.985**	**1.000**	0.500
	M4	0.922	0.912	0.870

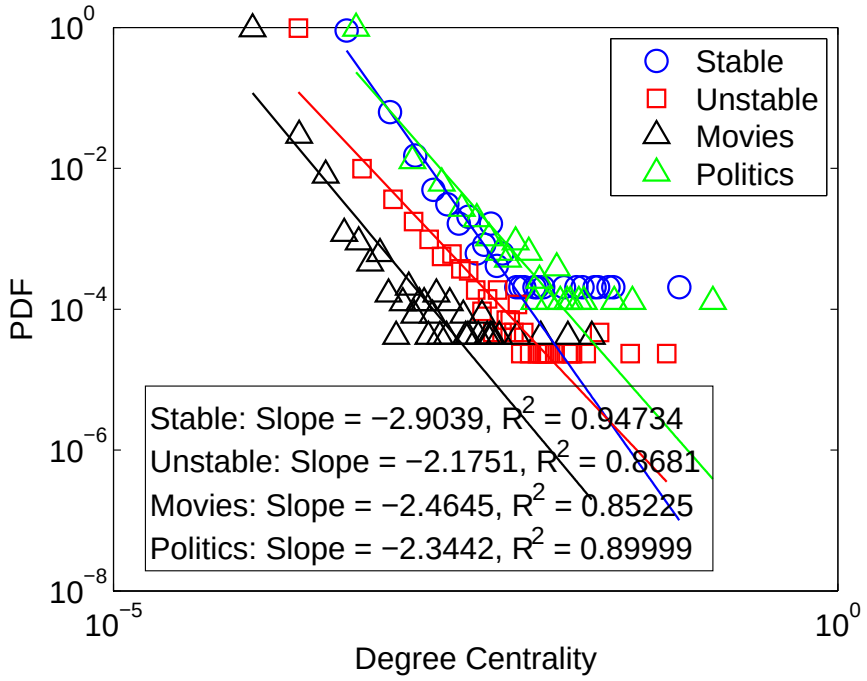


(a) Edge count

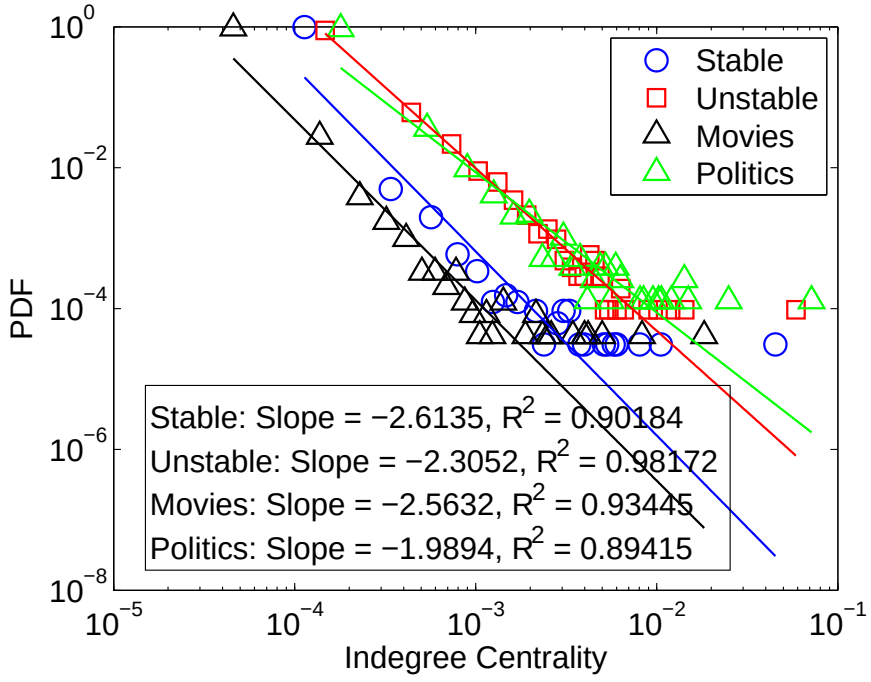


(b) Diffusion Lifetime

Figure 6.1: Examples of feature computation from the distribution of properties in individual tweet analysis method (M1).

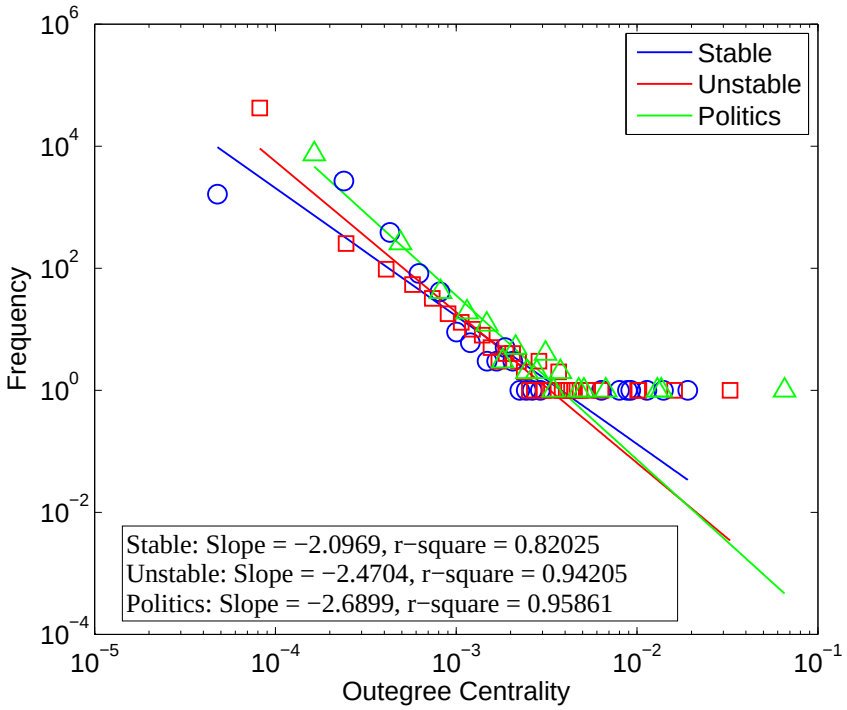


(a) Node Degree

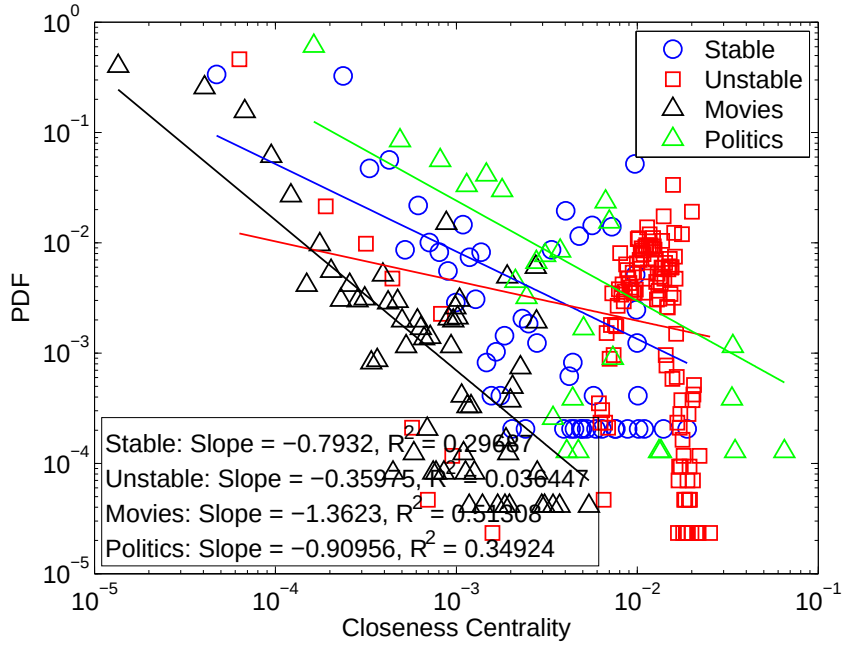


(b) Node In-Degree

Figure 6.2: Feature computation from the distribution of properties in multiple tweet static analysis method (M2).

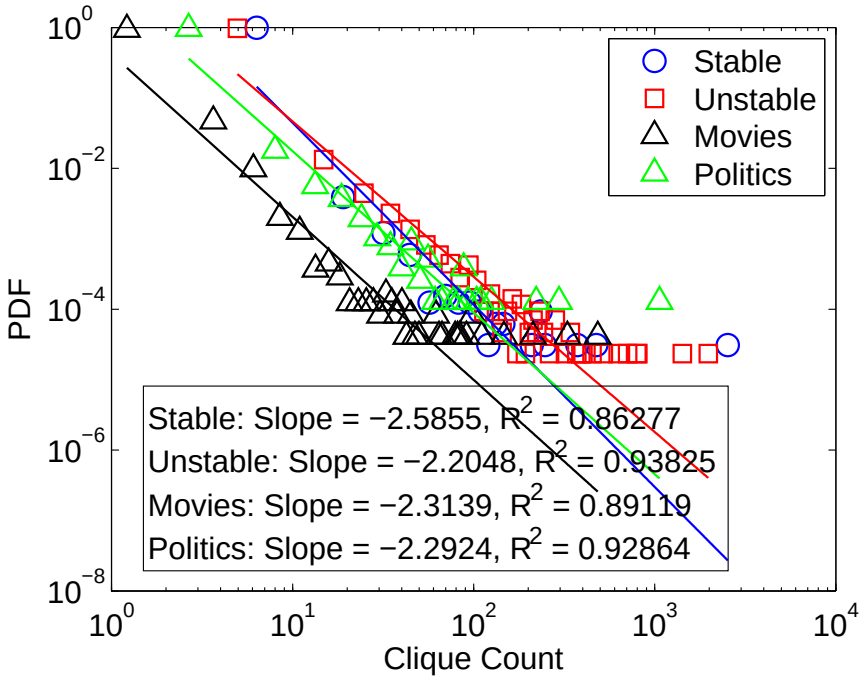


(a) Node Out-Degree

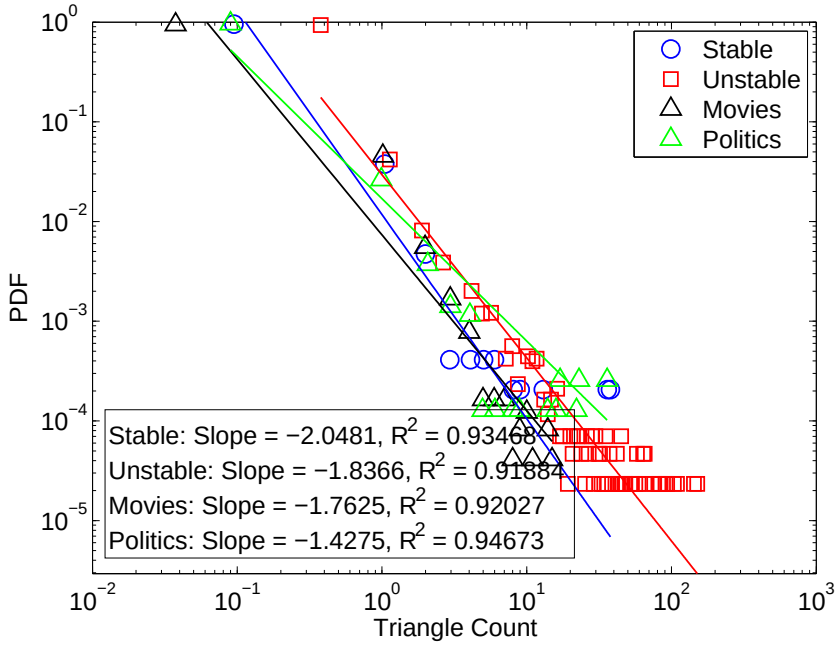


(b) Closeness Centrality

Figure 6.2: Feature computation from the distribution of properties in multiple tweet static analysis method (M2).



(a) Clique Count



(b) Triangle Count

Figure 6.2: Feature computation from the distribution of properties in multiple tweet static analysis method (M2).

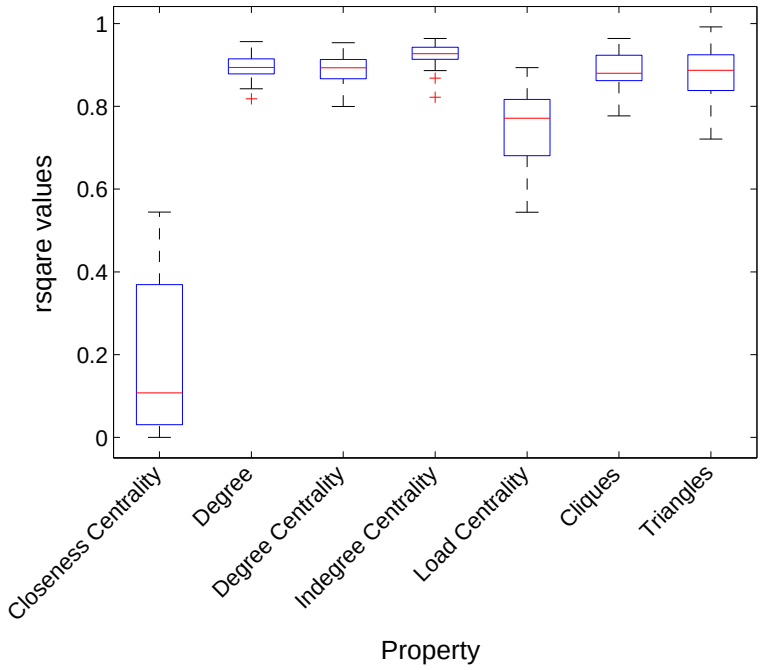
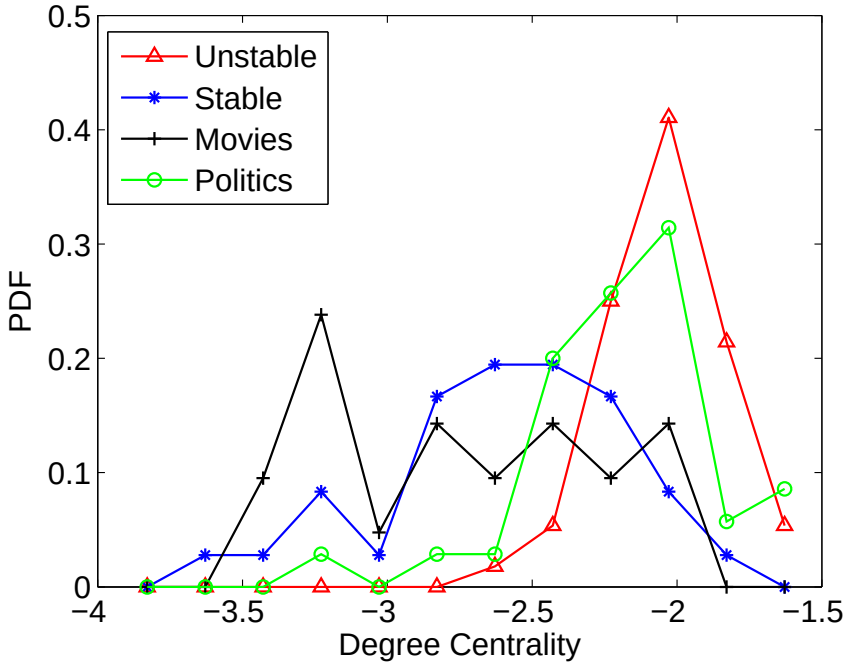
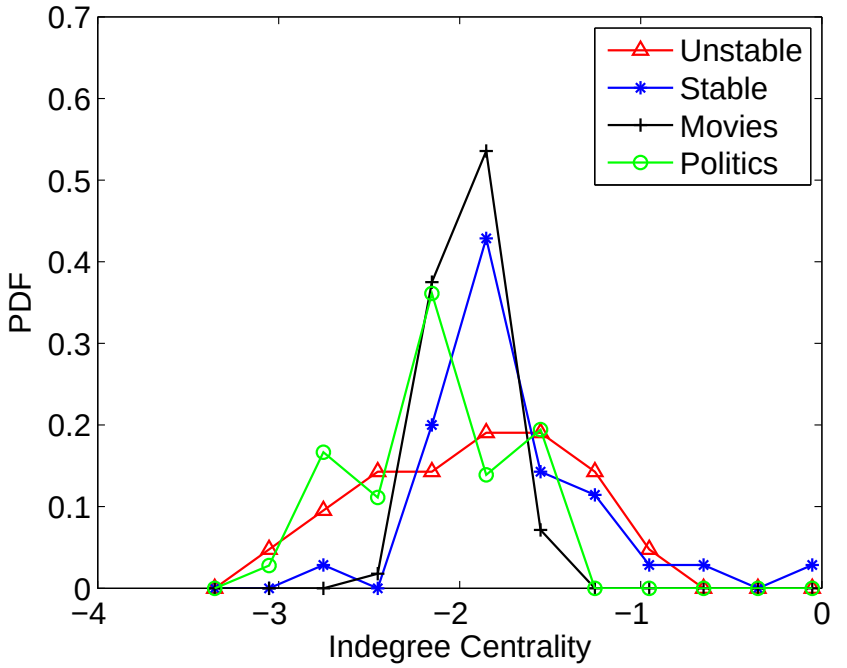


Figure 6.3: R-squared values for multiple tweets static graph properties distribution fitted to a linear equaion.

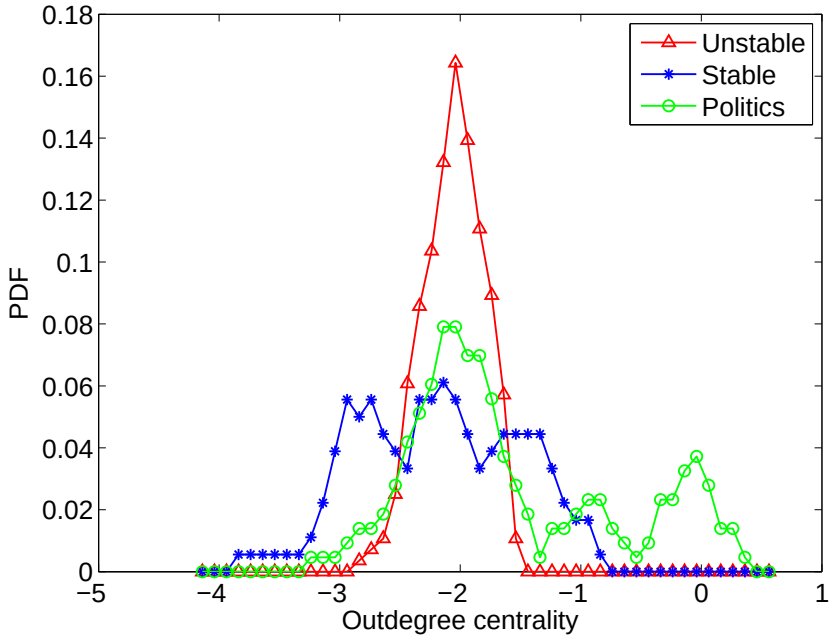


(a) Node Degree

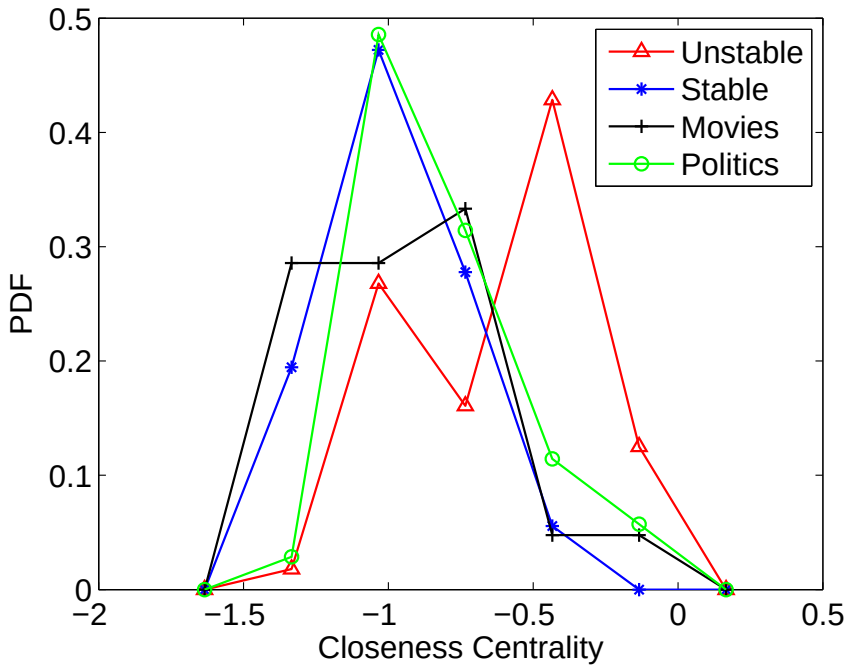


(b) Node In-Degree

Figure 6.4: Feature distributions from the distribution of properties in multiple tweet static analysis method (M2).

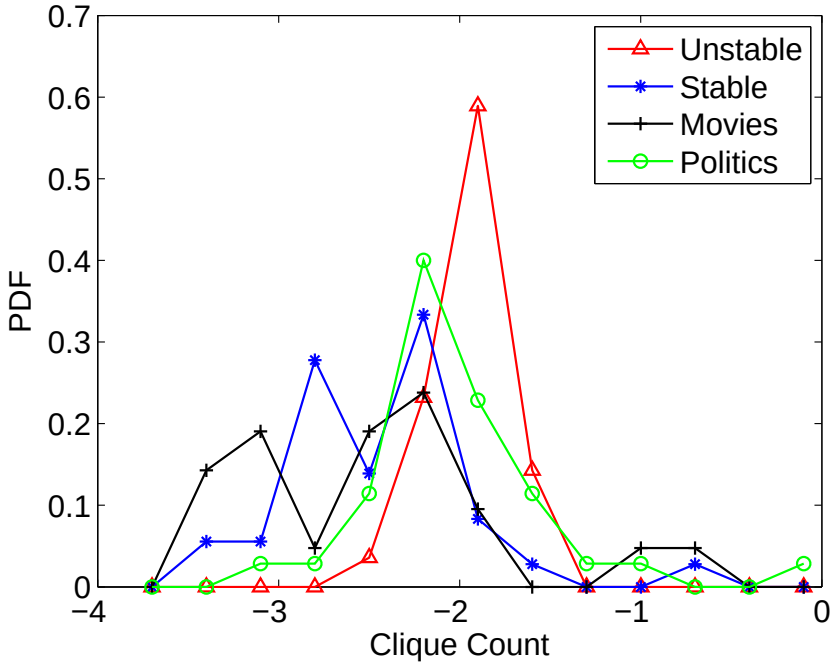


(a) Node Out-Degree

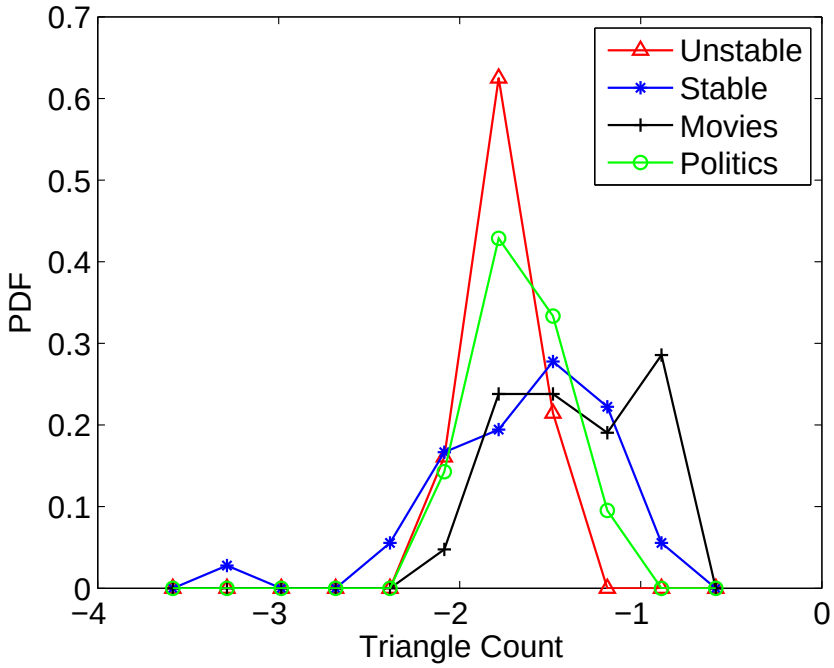


(b) Closeness Centrality

Figure 6.4: Feature distributions from the distribution of properties in multiple tweet static analysis method (M2).

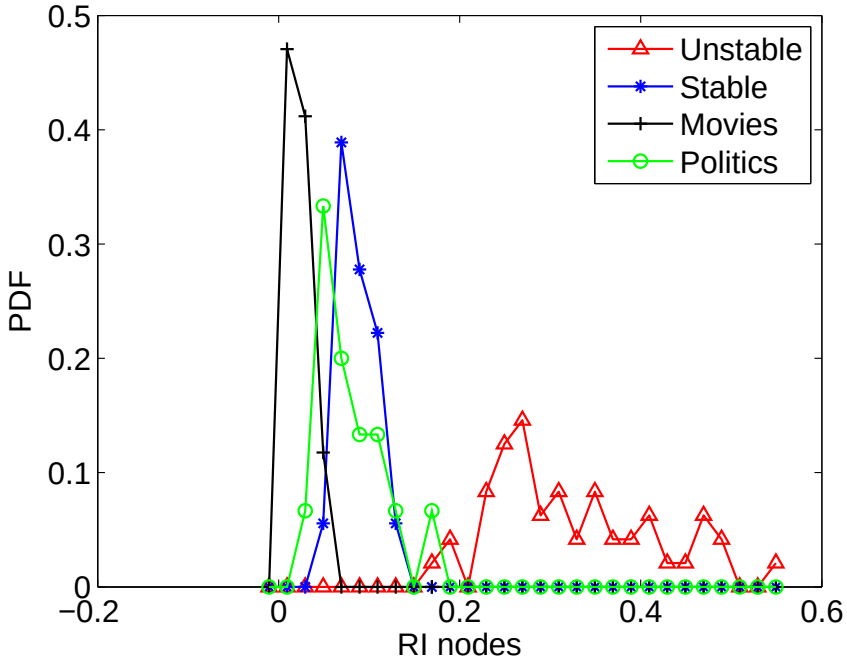


(a) Clique Count

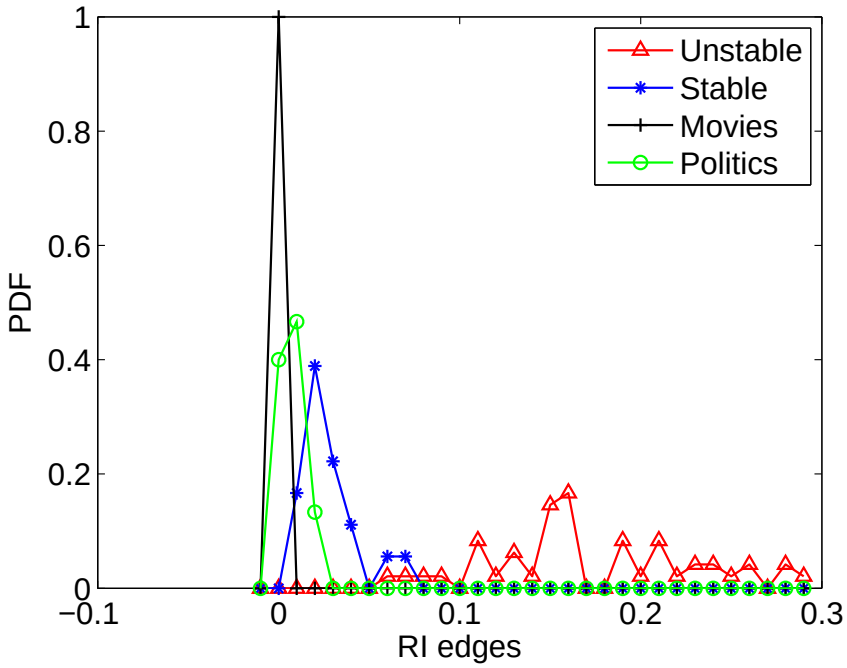


(b) Triangle Count

Figure 6.4: Feature distributions from the distribution of properties in multiple tweet static analysis method (M2).

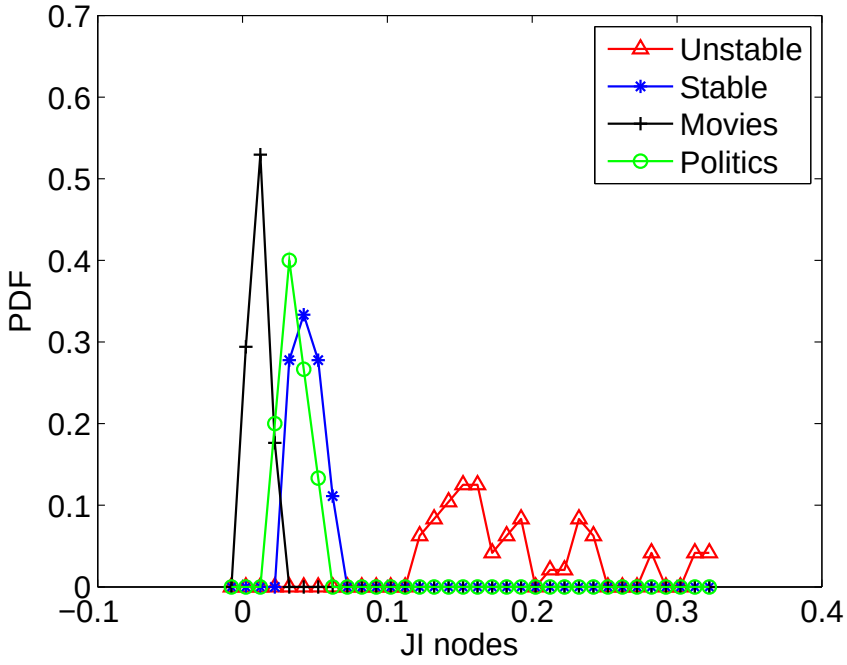


(a) Relative Inclusion for Nodes

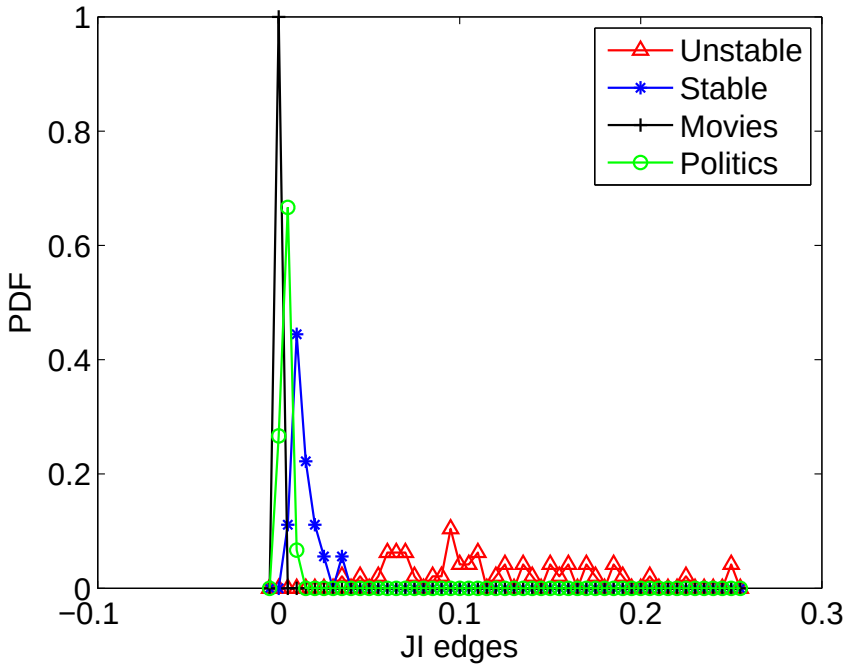


(b) Relative Inclusion for Edges

Figure 6.5: Feature distributions from the distribution of properties in multiple tweet dynamic analysis method (M3).

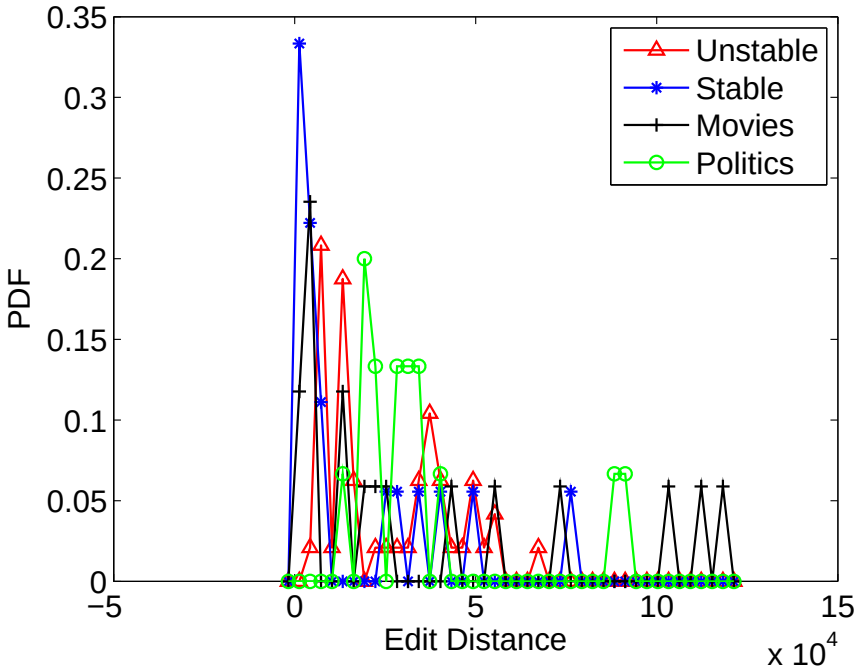


(a) Jaccard Index for Nodes



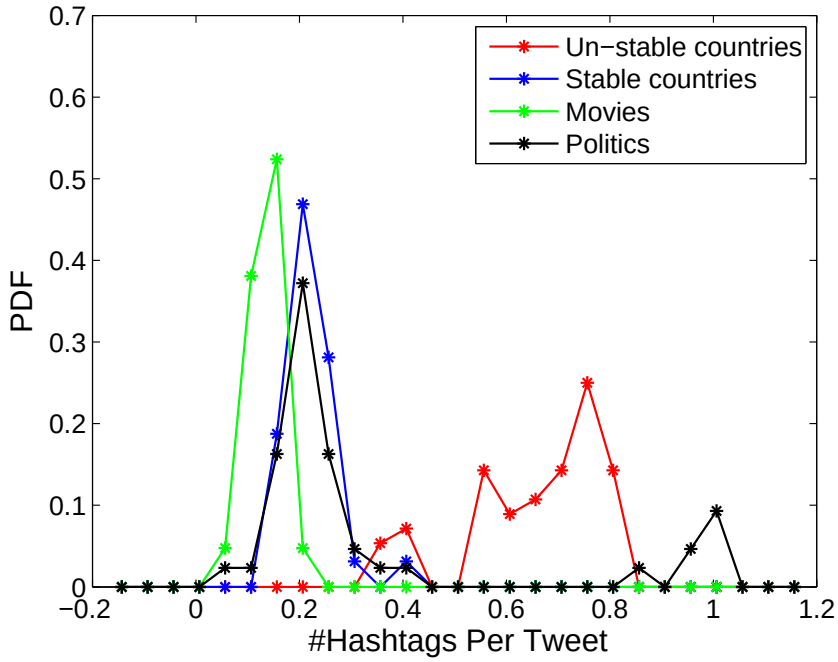
(b) Jaccard Index for Nodes

Figure 6.5: Feature distributions from the distribution of properties in multiple tweet dynamic analysis method (M3).

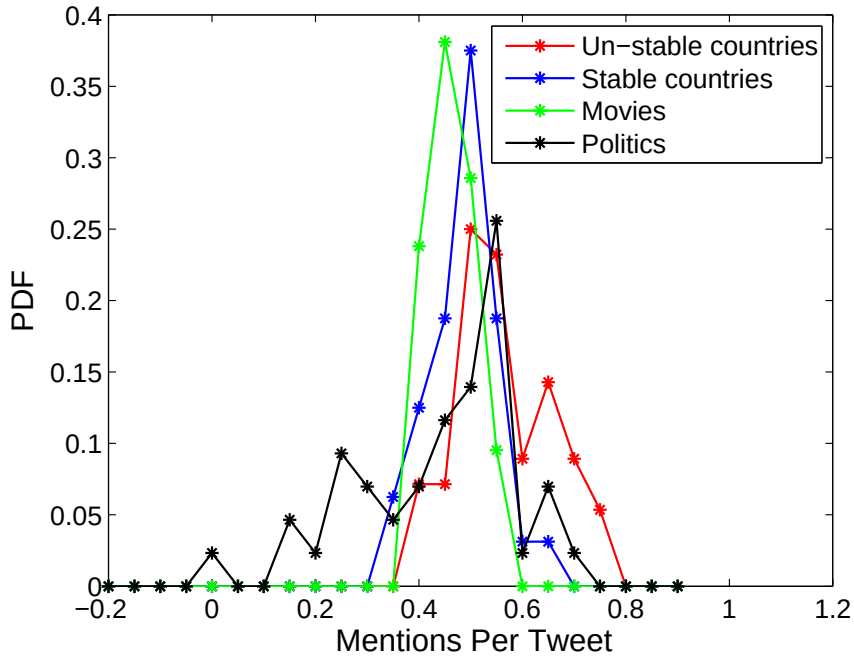


(a) Graph Edit Distance

Figure 6.5: Feature distributions from the distribution of properties in multiple tweet dynamic analysis method (M3).

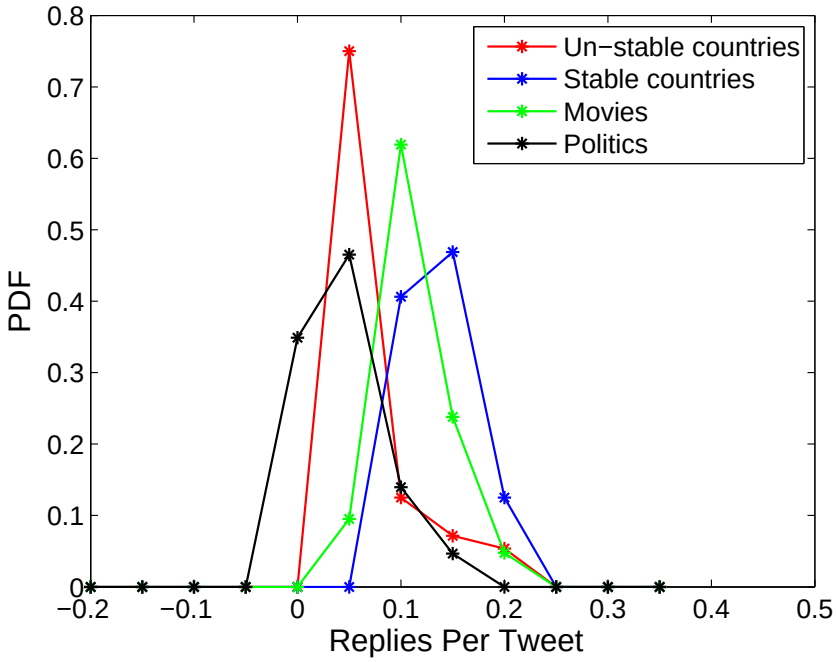


(a) Hashtags Per Tweet

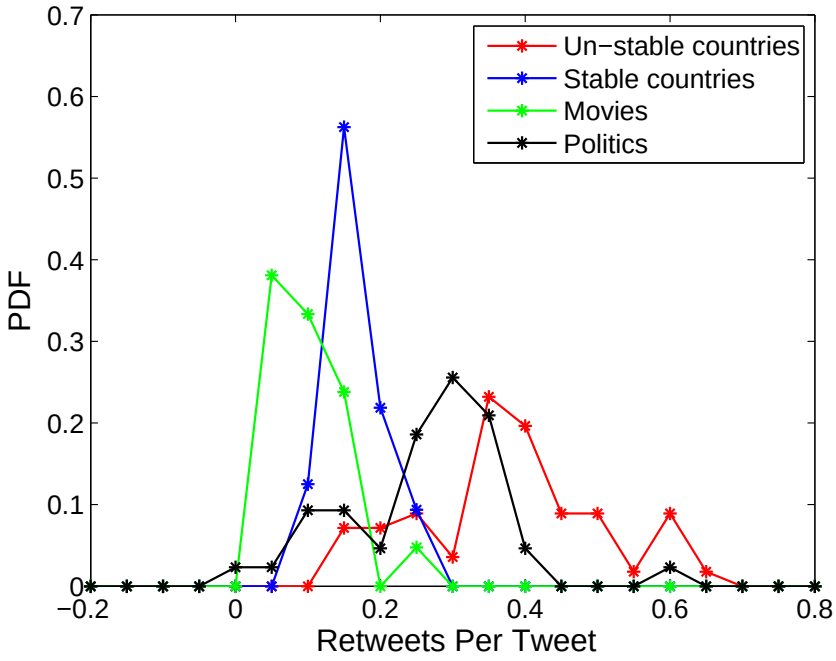


(b) Mentions Per Tweet

Figure 6.6: Feature distributions from the distribution of properties in tweet content analysis method (M4).

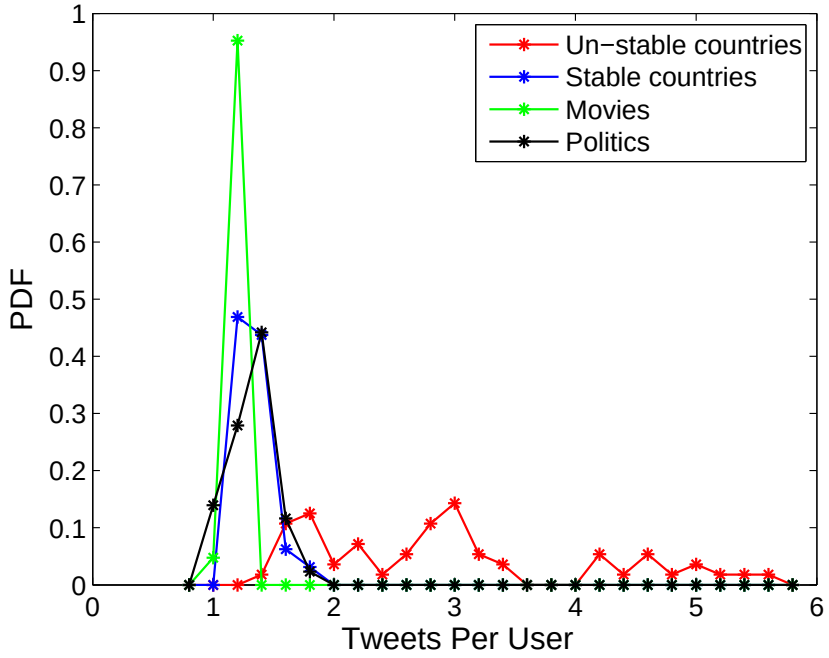


(a) Replies Per Tweet

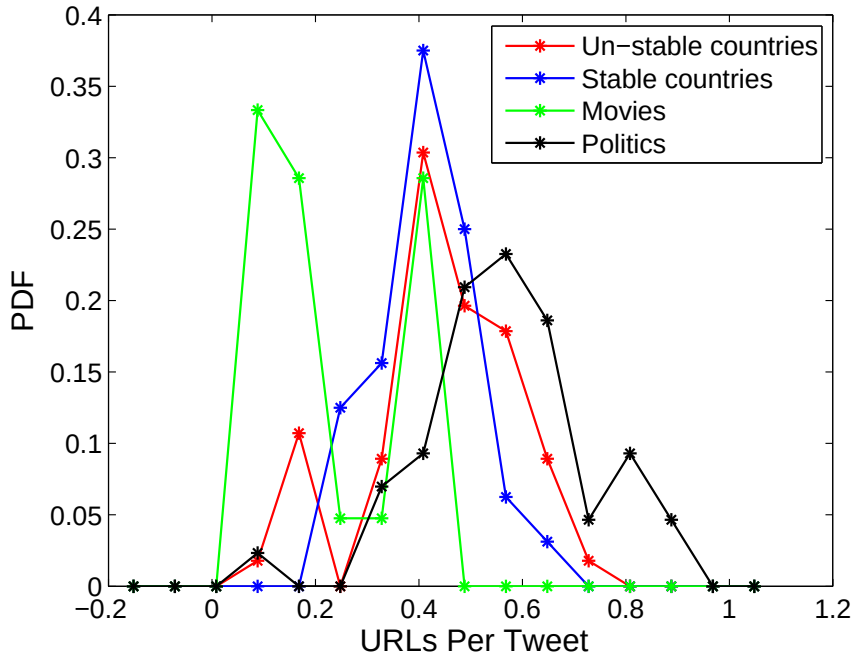


(b) Retweets Per User

Figure 6.6: Feature distributions from the distribution of properties in tweet content analysis method (M4).



(a) Tweets Per User



(b) URLs Per User

Figure 6.6: Feature distributions from the distribution of properties in tweet content analysis method (M4).

Chapter 7

Conclusions

In this chapter we will draw conclusions from the analysis provided in the thesis and we will also highlight the future directions regarding unrest detection in country using Twitter traffic.

7.1 Conclusion

In this thesis, we collected multiple Twitter data sets for countries experiencing political instability, countries that are stable, and other data sets pertaining to politics and movies. Our goal was to detect unstable countries using only properties of information cascade graphs. Towards this end, we computed a wide range of features for information cascades constructed using three different methods. For baseline comparison, we also computed content based features for classification. We used the extracted features with standard machine learning algorithms for automated classification. Based on our analysis we conclude following results for unrest detection:

1. The properties used for analysis of *diffusion of individual tweet* were not helpful for unrest detection and hence are not recommended for unrest detection task.

2. Structural graph properties used for *multiple tweet static graphs* performed well for detection of unrest when data was collected only for countries. But with the inclusion of data related to political campaigns the performance of this method degraded considerably, therefore using static interaction graphs is also not a viable option for unrest detection

problem.

3. And finally, the results of our experiments showed that analyzing *dynamic properties of information cascades of multiple tweets* provides the best classification performance. On the basis of this finding it is recommended to use this method for the purpose of unrest detection in a country.

In terms of classification algorithms, naïve Bayes consistently outperformed others in terms of accuracy. It achieved TP rate, FP rate, precision, and accuracy of 100%, 3%, 98% and 98.5% respectively, when applied to features extracted using multiple tweet dynamic analysis method.

7.2 Future Directions

In this thesis we have empirically proved the differences in Twitter traffic as a result of disturbance in the socio-political setup of a country. Using dynamic properties of information cascades of multiple tweets, we were able to detect the unrest in country with very good accuracy. The step forward from here would be to develop a method to exploit these differences in twitter traffic to forecast the socio-political unrest in any target country.

Bibliography

[1] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Effects of user similarity in social media. 2012.

[2] L. Backstrom and J. Kleinberg. Network bucket testing. In *Proceedings of the 20th international conference on World wide web*, pages 615–624. ACM, 2011.

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

[4] C. Bishop and S. S. en ligne). *Pattern recognition and machine learning*, volume 4. springer New York, 2006.

[5] J. Cheng, D. Romero, B. Meeder, and J. Kleinberg. Predicting reciprocity in social networks.

[6] F. Chierichetti, J. Kleinberg, and D. Liben-Nowell. Reconstructing patterns of information diffusion from incomplete observations.

[7] M. Conover, B. Gonçalves, J. Ratkiewicz, A. Flammini, and F. Menczer. Predicting the political alignment of twitter users. *Under Review*, 2011.

[8] M. Conover, J. Ratkiewicz, M. Francisco, B. Goncalves, A. Flammini, and F. Menczer. Political polarization on twitter. In *Proc. 5th Intl. Conference on Weblogs and Social Media*, 2011.

[9] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[10] T. Cover and J. Thomas. *Elements of Information Theory*, volume 6. Wiley, 1991.

[11] C. Delany. How social media accelerated Tunisia's revolution: An inside view, February 10 2011.

[12] W. Galuba, K. Aberer, D. Chakraborty, Z. Despotovic, and W. Kellerer. Outtweeting the twitterers-predicting information cascades in microblogs. In *Proceedings of the 3rd conference on Online social networks*, pages 3–3. USENIX Association, 2010.

[13] L. Gong, C. Teng, A. Livne, C. Brunetti, and L. Adamic. Coevolution of network structure and content. *Arxiv preprint arXiv:1107.5543*, 2011.

[14] http://pypi.python.org/pypi/tweetstream. tweetstream 1.1.1: Python package index, November 1 2011.

[15] http://twitter.com/. Twitter, November 1 2011.

[16] http://www.python.org/. Python programming language – official website, November 1 2011.

[17] D. Huffaker, C. Teng, M. Simmons, L. Gong, and L. Adamic. Group membership and diffusion in virtual worlds. *IEEE11*, 2011.

[18] M. Iliofotou, M. Faloutsos, and M. Mitzenmacher. Exploiting dynamicity in graph-based traffic analysis: techniques and applications. In *Proceedings of the 5th ACM Conference on Emerging Networking Experiments and Technologies*, pages 241–252, 2009.

[19] U. Karkada, L. Adamic, J. Kahn, and T. Iwashyna. Limiting the spread of highly resistant hospital-acquired microorganisms via critical care transfers: a simulation study. *Intensive care medicine*, pages 1–8, 2011.

[20] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th ACM Conference on World Wide Web*, pages 591–600, 2010.

[21] K. Lerman and R. Ghosh. Information contagion: An empirical study of the spread of news on digg and twitter social networks. In *Proceedings of 4th International Conference on Weblogs and Social Media (ICWSM)*, 2010.

[22] A. Livne, M. Simmons, E. Adar, and L. Adamic. The party is over here: Structure and content in the 2010 election. In *Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[23] D. Romero, B. Meeder, and J. Kleinberg. Differences in the mechanics of

information diffusion across topics: Idioms, political hashtags, and complex contagion on twitter. In *Proceedings of the 20th ACM Conference on World Wide Web*, pages 695–704, 2011.

[24] G. Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, 1966.

[25] E. Sadikov, M. Medina, J. Leskovec, and H. Garcia-Molina. Correcting for missing data in information cascades. In *Proceedings of the fourth ACM Conference on Web Search and Data Mining*, pages 55–64, 2011.

[26] X. Wei, J. Yang, L. Adamic, R. de Araújo, and M. Rekhi. Diffusion dynamics of games on online social networks. In *Proceedings of the 3rd conference on Online social networks*, pages 2–2. USENIX Association, 2010.

[27] D. West et al. *Introduction to graph theory*, volume 2. Prentice Hall Upper Saddle River, NJ.:, 2001.

[28] C. Wilson, B. Boe, A. Sala, K. Puttaswamy, and B. Zhao. User interactions in social networks and their implications. In *Proceedings of the 4th ACM European Conference on Computer Systems*, pages 205–218, 2009.

[29] I. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

[30] S. Wu, J. Hofman, W. Mason, and D. Watts. Who says what to whom on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 705–714. ACM, 2011.

[31] S. Wu, C. Tan, J. Kleinberg, and M. Macy. Does bad news go away faster? In *In Proceedings of the International Conference on Weblogs and Social (ICWSM)*, 2011.

[32] S. Yardi and D. Boyd. Dynamic debates: An analysis of group polarization over time on twitter. *Bulletin of Science, Technology & Society*, 30(5):316, 2010.