MAKING SENSE OF SOCIAL MEDIA TEXT AND THE SPREAD OF RUMOURS IN ONLINE SOCIAL NETWORKS — AN INTERDISCIPLINARY APPROACH

by

Anh Dang

Submitted in partial fulfillment of the requirements
for the thesis defense of Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
July 2019

Dedicated to my family

Table of Contents

List of Tables

List of Figures

Abstract

As the spread of rumours in online social networks (OSNs) has grown at an alarming pace, there is a growing need to better understand the social and technological processes behind this trend. This research proposes an interdisciplinary approach to study the effects of rumours in OSNs, with the end goal of developing and validating a set of interactive visualization models that will help researchers as well as members of various OSNs to detect and prevent the rapid spread of rumours in these networks. The strength of the proposed research is that it adopts an interdisciplinary approach to study the phenomenon by integrating valuable insights from different established disciplines, such as sociology, psychology, information science, and computer science, to create a holistic view and understanding of how rumours are spread in OSNs.

The thesis first studies the impact of short and noisy nature of social media text, which could significantly affect the performance of Natural Language Processing (NLP) systems. We introduce a new terabyte-scale corpus that is created from Reddit comments from Oct 2007 to Aug 2016 and propose a novel approach to compute the semantic similarity between social media texts. The proposed semantic similarity algorithm will alleviate the inherent limitation of social media texts and improve the results of NLP systems using social media data. Then, we propose a visual framework to detect and cluster memes in OSNs. Our algorithms could conclusively identify the emerging and trending memes in OSNs. After discovering memes, we propose a visualization framework for collecting, analyzing, and visualizing memes and rumours in OSNs using theories rooted in psychology, sociology, information science, and computer science. This framework allows end users to collect data about a specific rumour and see its spread pattern, topics over time, sentiment analysis, and user interaction graphs. Using established psychological theories, we classify users based on how they interact in a rumour. Finally, we try to detect the truth of rumours based on selective feature sets that are derived from the proposed visualization tool and established social science theories.

List of Abbreviations Used

ApEn	Approximate Entropy
GTM	Google Tri-gram Method
ICW	Internal Centrality-Based Weighting
JC	Jaccard Coefficient
LSA	Latent Semantic Analysis
LDA	Latent Dirichlet Allocation
MRPC	Microsoft Research Paraphrase Corpus
NLP	Natural Language Processing
OSNs	Online Social Networks
PI	Paraphrase Identification
SVD	Singular Value Decomposition
SS	Semantic Similarity
SJC	Semantic Jaccard Coefficient
SSR	Similarity Score Reweighting with Revelance User Feedback
TLSA	Topic-based Latent Semantic Analysis
TF-IDF	Term Frequency, Inverted Document Frequency
VA	Visual Analytics
WTMF	Weighted Textual Matrix Factorization
XML	Extensible Markup Language

Acknowledgements

I want to express my profound gratefulness to my supervisor, Dr. Evangelos Milios and Dr. Michael Smit, for their continuous support during my Ph.D. study. I am grateful for the opportunity to study, learn, and exchange research ideas in the Malnis group. During my study, with my supervisors' support, I have the chance to travel and present my research with other research scholars around the world.

My sincere thanks and appreciation goes to Dr. Abidalrahman Moh'd, Dr. Aminul Islam, and Dr. Rosane Minghim. The completion of my work would not have been possible without their valuable suggestions from detecting very slight writing style errors to suggesting innovative ideas for my publications.

I want to thank you to Dr. Vlado Keselj and Dr. Fernando Paulovich for being members of the examining committee. I want to thank Dr. Diana Inkpen to act as my Ph.D. thesis external examiner and Dr. William Baker as my IDPHD departmental thesis chair.

Finally, I would like to thank my parents and my wife, Yen Le, for their support and encouragement during my study.

Chapter 1

Introduction

1.1 Motivation for Research

Online social networks (OSNs) are networks of online interactions and relationships that are created and maintained through various social networking sites such as Facebook, LinkedIn, Reddit, and Twitter. Recently, millions of people and organizations have used OSNs to share information and connect with friends and strangers. OSNs have been especially useful for disseminating information in the context of political campaigning, news reporting, marketing, and entertainment [95].

As the amount of social media is increasing every day, researchers have started to take advantage of these massive amounts of data for day to day research. For example, researchers have tried to study the impact of social media on 2016 US elections [6], the impact of social media on HIV research [147], and the effect of social media data on NLP systems [162]. Although social media data has many intrinsic advantages, such as a large amount of data, the up-to-date data and a large number of user bases, it also has many disadvantages. One major challenge is that social media data is intrinsically short and noisy. Wang et al. [153] showed that text corpora have diverse properties and affect the performance in many NLP applications. More importantly, we observe that no existing large-scale n-gram corpus is created specifically from social media text. This has motivated us to create an n-gram corpus that is derived from 1.65 billion comments in the Reddit corpus [12] and to make it available to the research community. Using the intrinsic characteristics of this large-scale corpus, we introduce a semantic similarity and paraphrase identification algorithm that is designed specifically for social media texts [47].

This growing usage of social media has created both challenges and opportunities. Recently, end users have relied on OSNs to learn more about breaking news stories, trending topics, or memes (a meme is a unit of information that can be passed from person to person in OSNs [87]). Those stories could be true or false. As well as spreading credible information, OSNs can also spread rumours. For example, so many rumours were disseminated via OSNs during the Swine Flu outbreak in 2009 that the US government had to tackle it officially on their website [119, 60]. This raises the question of how to identify and prevent the spread of rumours in OSNs. One of the first steps of detecting rumours is to detect the spread of memes (stories) and determine their veracity. In this thesis, we try to detect the spread of memes using semantic similarity, clustering, visualization, and user feedback. Our end goal is to detect the spread of memes in real time so that we could analyze those memes and try to detect if they are true or false.

OSNs can not only spread reliable information, but they can also spread misleading information. A rumor is defined as a statement whose true value is not easy to verify, and it appears and is disseminated in uncertain situations [137]. Rumors are spread when the rumor topics are of interest to a large number of individuals, and their truths are not easy to verify [56]. Online Social Networks (OSNs) have recently emerged as the favored means for both spreading credible information [95] and rumors [81]. Existing research tries to detect and categorize rumours in OSNs by applying different feature sets using a supervised machine learning approach without considering the morphing characteristics of rumors. In addition, rumor characteristics could be different in various sources (e.g., Twitter vs. Reddit) or categories (e.g., politic vs. non-politic rumors). This makes detecting rumors a challenging task unless the phenomenon is examined more deeply. In this research, we proposed a visualization framework that could collect, analyze, and visualize rumour spread in OSNs. The proposed framework adopts knowledge from various disciplines, such as sociology, psychology, information science, and computer science to provide a complete view of how rumors are spread and how users interact with each other inside a rumor. As there are various forms of rumors in OSNs (e.g., fake news is a form of deliberate false rumors and is spread and published as authentic news [156]), in this thesis, we focus on studying the spread of rumors in general and how to detect and debunk them.

Researchers have found that false rumors, hoaxes, or fake news (another form of false rumors) are more likely to be more popular and spread further than true rumours [115]. Those rumors have a detrimental effect on an individual's reputation or societies. For example, during the US 2016 election, a large number of voter population had seen fake news and believed in those false stories [6]. Another example is that rumors could play a detrimental effect on the stock markets [94]. Detecting rumor veracity and preventing its spread in the early stage in OSNs is an essential step for end users to make a better-informed decision-making process. Recently, OSNs, such as Facebook and Google, have partnered with Snopes.com [144] and Politifact.com [130] to validate and debunk rumor stories in OSNs. This approach uses human knowledge to manually categorize if a rumor is “False”, “Mostly False”, “True”, “Mostly True”, and “Half True”. Manually labeling all those rumors is a time-consuming task in real time. OSNs could effectively be a useful source of human input to debunk rumors [48]. We propose a newly-created rumor dataset with finer-grained truth levels (according to Snopes.com and Politifact.com) and use this dataset to study how early we could effectively identify the truth of rumors.

The primary goal of my research is to collect, study, analyze, visualize, detect, and limit the spread of rumours in OSNs. To achieve this goal, the proposed research will study and apply relevant knowledge from multiple disciplines including sociology, psychology, information science, and computer science.

1.2 General Background

1.2.1 Studying the Intrinsic Shortness and Noisiness of Social Media Text

Corpus-based machine learning algorithms have an advantage over knowledge-based ones as they do not involve in a human which can be expensive. The Google web 1T n-gram corpus [22] has been used for text relatedness [90] and linguistic steganography [29]. Google Book n-gram corpus [116] has been used to study the changing psychology of culture [71], concepts of happiness [125], and mapping book to time [89]. Twitter n-gram corpus [80] only provides a small subset of social media n-grams in Twitter. As Twitter does not share full texts of Tweets as a large corpus, it is not feasible to collect, create and share terabyte-scale n-gram corpus for Tweets. Unlike Twitter, Reddit data is open, and users can query all contents from the website. Although OSNs have been used intensively for research in recent years, there is no existing corpus that could be shared and provide insights from massive social network text. To the best of our knowledge, this new corpus is the first large-scale n-gram corpus that provides n-grams with a temporal feature (monthly) that is designed specifically for massive user-generated social media text.

1.2.2 Studying and Visualizing Rumours

The idea behind visualization is that it will allow users without any knowledge to quickly analyze, collect, and visualize rumours. Getting insights from rumours will help to better understand and detect rumours. Ratkiewics et al. [136] proposed a data mining framework using Twitter social data and sentiment analysis to detect the spread of political misinformation in the 2010 U.S. midterm elections. Qazvinian et al. [132] introduced a framework that combines statistical models with some natural language processing to identify deceptive messages on Twitter. Budak et al. [24] applied a similar concept of inoculation in treating the spread of epidemics disease by introducing the notion of good campaigns, for example, from official announcements to fight against misinformation dissemination and reduce the number of users affected by misinformation propagation. There are also some early, industry-led solutions to this problem such as Veri.ly, an online platform to verify and evaluate the credibility of information based on crowd-sourced evidence, or PolitiFact.com, a website used by reporters and editors to evaluate statements from political parties based on fact-checking. Most of the recent research [136, 132, 24, 123] did not apply or applied only limited rumour characteristics information, such as topics, sentiment analysis, and social science theories. Although these visualization applications [134, 143] are very useful tools, they do not allow many interactions between users and their systems. The proposed visualization framework provides a robust visual analytic visualization tool so that end users could collect, analyze, and explore to better understand and associate different aspects of rumors.

1.2.3 Detecting Rumors in Online Social Networks

The first step in detecting a rumour is to identify the emerging memes in OSNs. A meme is a unit of information that can be transmitted from users to users in OSNs [87]. Cataldi et al. [26] proposed an approach that monitored the real-time spread of emerging memes in Twitter. The authors defined an emerging term as one whose appearance frequency had risen within a short period and had not emerged or was only rarely discussed in the past. A navigable topic graph is constructed to connect semantically related emerging terms. Emerging memes are extracted from this graph based on semantic relationships between terms over a specified time interval. Leskovec et al. [100] proposed a meme-tracking framework to monitor memes that travel through the Web in real-time. The framework studied the signature path and topic of each meme by grouping similar short and distinctive phrases together. Our proposed approach is different from the existing work in that it adopts the use of semantic similarity measures and Wikipedia concepts to detect memes.

The first publicly available rumor dataset is provided by Qazvinian et al. [132]. This dataset includes 10,000 tweets involving five different rumors. Each tweet is annotated as "related" or "unrelated" to a rumor. A dataset of 100 million tweets involving 72 rumors

(41 true and 31 false) was constructed by Giasemidis et al. [67] and a machine learning approach was applied to it to classify whether those rumors are true or false. The PHEME dataset includes 1,972 rumorous and 3,830 non-rumorous tweets about five breaking news stories [51]. The dataset provided by Kwon et al. [96] is a collection of tweets for 61 rumors and 51 non-rumors, used to study how various feature sets affect the accuracy of rumor detection over time. As most of the existing datasets only include two rumour veracity categories ("false" and "true"), our research aims to provide a rumor dataset that could be used to identify the truthfulness of rumors in one of the five categories: "False", "Mostly False", "True", "Mostly True", "Half True". These fine-grained truth levels are used to reflect the nature of rumor spread in OSNs.

To date, most of the work in this emerging area has been conducted in Computer Science. However, in order to develop effective methods for rumour detection and prevention in OSNs, we first need to understand who spreads rumours online, why, and how. Thus, only by using an interdisciplinary approach can we succeed in addressing this research challenge.

1.3 Bridging the Gap — An Interdisciplinary Approach

Since the study of OSNs and rumours is inherently interdisciplinary, finding the answers for the research questions in this proposal cannot be adequately addressed from a single discipline. Therefore, incorporating methods, theories, and results from different disciplines, such as Psychology, Sociology, Information Science, and Computer Science will provide a more comprehensive picture of how rumours are spread in OSNs.

1.3.1 Information Science

Information-seeking behaviour models describe a way to gather information for specific needs. Zipf [167] used the "principle of least effort" to explain how people tend to use the most available tools to seek information and that the process of information-seeking is stopped when a threshold of minimally acceptable results is passed. Dervin [53] used the concept of sense-making to describe how people use common sense and external information to find answers for uncertain situations.

Information Diffusion models are another group of models in Information Science that are highly relevant to this proposal. They focus on how information is disseminated among online participants. Dotey et al. [145] defined the theory of information cascades, which is the behaviour of people making decisions based on the influence of others in OSNs. Studying these models will help to find users who play the most important roles in the process of spreading rumours in OSNs. Investigating and integrating these models [38, 145] into the proposal will help to study the role of information-seeking behaviour in trying to explain why certain types of rumours are more likely to be disseminated than others.

1.3.2 Sociology

Sociologists play an essential role in studying social networks in general and online social networks in particular. A number of sociological theories and concepts are relevant to the proposed research.

The Strength of Weak Ties: Granovetter [70] argued that information flows more effectively between people in social networks through weak connections (connections with a friend we do not know well) than through strong links (close friends).

The Threshold Model: Granovetter [69] stated that the behavior of individuals in social networks depends on an endless number of other people doing the same behaviour in the same context. Also, each individual will have a value for that constant number, called a threshold value. The threshold value is different for each individual and depends on social status, education, age, and personality.

Homophily suggests that people with similar characteristics, such as gender, race, or ethnicity, are more likely to be connected on social networks.

Studying the connections between those models, theories, concepts and the spread of rumors in OSNs will help find the answers to how to effectively identify influencing users in rumor spread and use this information to limit the spread of rumors and debunk them.

1.3.3 Psychology

The study of how people process information and modify their existing knowledge is critical in explaining why some rumours are more effectively spread than others. Several psychological concepts are highly related to this research.

Crowd manipulation is a way to manipulate the behaviour of a crowd towards a specific end. This technique has been widely used in political campaigns to spread misleading information to deter voters from voting for an opponent.

Herd Mentality describes how people behave and act similarly to the majority of those around them. Muchnik et al. [120] showed that the so-called herd mentality also affects how people behave and share information in OSNs. For example, people will tend to share rumours in OSNs if this information is shared by most of their connections.

Studying why people spread rumours and how users interact with each other inside a rumour may help us to identify the early stage of rumour spreading in OSNs and explain why some rumours usually go viral and become unmanageable in OSNs. In this research, we adopt two psychological rumor spread theories [20, 56] to study the underlying reasons why users spread rumors in OSNs.

1.3.4 Computer Science

In this research, the proposed visualization framework will use knowledge from some established areas, such as machine learning, data mining, and information visualization, to gather, manipulate, and analyze unstructured data from OSNs. With these processed and structured data, we can further investigate the structure of OSNs and how rumours are spread between online users.

1.3.5 Goals and Objectives

The goals of this thesis are to:

- study the intrinsic nature of social media text and propose an approach to improve its noisiness and shortness.
- propose and improve the effectiveness of the meme detecting tasks in real-time using semantic similarity.
- collect, analyze, and visualize rumour spread in OSNs. Provide end users a valuable tool to investigate the characteristics of a rumour and compare those characteristics among various rumours. This visualization framework integrates information diffusion models, social science theories, social network analysis, sentiment analysis, and text mining techniques to facilitate data exploration and analysis for online rumor spread.
- detect and debunk rumours in OSNs using a crowd-sourcing approach and social science theories.

The final goal of this thesis is to provide a holistic view of studying, analyzing, collecting, detecting, visualizing and debunking rumor spread in OSNs. The proposed interdisciplinary approach is summarized in Figure 1.1.

1.4 Thesis Outline

Chapter 2 tackles the problem of noisiness and shortness of social media texts and is published in Dang et al. [47]. Chapter 3 solves the problems of detecting memes in social media and is published in Dang et al. [41, 43]. Chapter 4 provides a visualization framework for collecting, analyzing, and visualize rumours and is an extended version of our research article [44]. Chapter 5 tries to understand why people spread rumours in OSNs and is

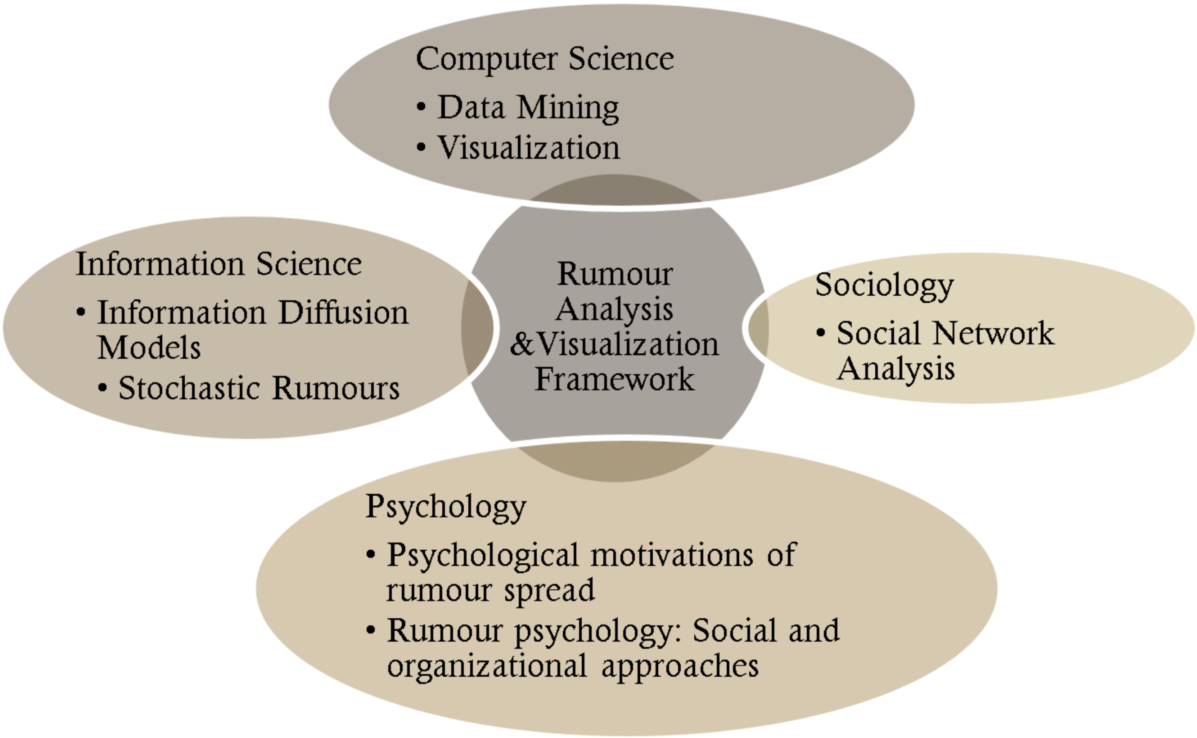

Computer Science
• Data Mining
• Visualization
Information Science
• Information Diffusion Models
• Stochastic Rumours
Rumour Analysis &Visualization Framework
Sociology
• Social Network Analysis
Psychology
• Psychological motivations of rumour spread
• Rumour psychology: Social and organizational approaches

Figure 1.1: The interdisciplinary nature of my thesis.

published in Dang et al. [45]. Finally, Chapter 6 tries to detect and verify the truth of a rumour and is published in Dang et al. [46].

Chapter 2

Reddit Temporal N-gram Corpus and its Applications on Paraphrase and Semantic Similarity in Social Media using a Topic-based Latent Semantic Analysis

This chapter introduces a new large-scale n-gram corpus that is created specifically from social media text. Two distinguishing characteristics of this corpus are its monthly temporal attribute and that it is created from 1.65 billion comments of user-generated text in Reddit. The usefulness of this corpus is exemplified and evaluated by a novel Topic-based Latent Semantic Analysis (TLSA) algorithm. The experimental results show that unsupervised TLSA outperforms all the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015: paraphrase and semantic similarity in Twitter tasks. The basis of this chapter is from the published paper [47].

2.1 Introduction

A word n-gram is a continuous sequence of n words from a corpus of texts or speech. Word n-gram language models are widely used in Natural Language Processing (NLP), such as speech recognition, machine translation, and information retrieval. The effectiveness of a word n-gram language model is highly dependent on the size and coverage of its training corpus [37]. A simple algorithm can outperform a more complicated algorithm if it uses a larger corpus [124]. Many large-scale corpora [22, 10, 154] based on web contents have been created for this purpose. As the use of social media is increasing, Online Social Networks (OSNs) have become a norm to spreading news, rumours, and social events [95]. This growing usage of social media has created both challenges and opportunities. One major challenge is that social media data is intrinsically short and noisy. A study by Wang et al. [153] revealed that different text corpora have significantly different properties and lead to varying performance in many NLP applications. More importantly, we observe that there is no existing large-scale n-gram corpus that is created specifically from social media text. This has motivated us to create an n-gram corpus that is derived from 1.65 billion comments in the Reddit corpus [12] and make it available to the research community. There are two main features of this corpus that do not exist in the available large-scale corpora in the literature: monthly time-varying (temporal) and purely social media text. This corpus will allow researchers to analyze and make sense of massive social network text, such as finding corresponding terms across time [164] and improving named entity recognition in tweets [102]. Moreover, a cloud-based visualization interface is implemented to allow end users to query any n-gram from the corpus.

Although there are many applications that can be derived from this corpus, in this chapter, we use the Paraphrase Identification (PI) and Semantic Similarity (SS) tasks of SEMEVAL 2015 [158] to exemplify the usefulness of this corpus. Paraphrases are words, phrases or sentences that have the same meaning, but their vocabulary may be different [158]. PI and SS tasks have a strong correlation, as both focus on the underlying structural and semantic similarity between two texts (e.g., “selfie” is a paraphrase of “picture of myself”). Improving the results of PI and SS helps to increase the performance of NLP systems, such as statistical machine translation [109] and plagiarism detection [11]. PI and SS have been studied intensively for formal text with important results as shown in Par [127]. As social media text is usually very short (e.g., 280-character limit for Twitter) and noisy (flexible nature of personal communication), many NLP systems suffer from the large degree of spelling, syntactic and semantic variants, for example, “ICYMI”= “In case you missed it” or “b/c I love u” = “Because I love you”. Traditional approaches have been studied intensively and proved not to work well for social media text [162]. A few preliminary results have shown that the shortness and noisiness of social media text have significantly decreased the performance of PI [162, 160] and SS tasks [73, 40]. In this chapter, we proposed a Topic-based Latent Semantic Analysis (TLSA) approach for the SS task, which assigns a semantic similarity score between two social media texts. Next, we use this similarity score to determine if two texts are a paraphrase of each other.

Latent Semantic Analysis (LSA) has been widely used for semantic text similarity tasks because of its simplicity and efficiency [97]. LSA has been used as a strong benchmark in the Microsoft Research sentence completion challenge [170] and its baseline has outperformed a few state-of-the-art neural network models [117]. However, LSA has its own drawbacks. Its models are trained on a large corpus where words in the same document have a stronger relationship. This does not consider how close two words are in a text (“apple” and “fruit” are closer in the 5-gram “apple is a fruit” instead of a whole document) [83]. Another example is two topics “Barack Obama” and “Hillary Clinton” have a different meaning in two contexts “2012 US presidential race” and “2016 US presidential race”. In the first one, they are opponents, while in the second one, “Barack Obama” endorsed “Hillary Clinton”. In addition, LSA is usually trained on a whole corpus. This makes it not scalable with an intrinsic, dynamic, and large-scale nature of social network data. To address this issue, we proposed an approach to train an LSA model that considers the topic being discussed. This proposed LSA model is trained on word 5-grams instead of whole documents. The proposed TLSA method achieved the best result for the SS task and is more scalable compared to other LSA models. Combining TLSA with sentiment analysis, the proposed approach also achieved the best result for PI task in SEMEVAL 2015. These are the contributions of our paper:

- We create a new word n-gram (1-5) social network corpus from 1.65 billion comments of Reddit[1]. This corpus has two distinctive characteristics that are useful for social media applications: temporal and large-scale social media text.
- We implement a cloud-based visualization interface so that end users can query and analyze the social media n-grams in real time.
- We propose TLSA[2], a Topic-based Latent Semantic Analysis model that is trained on word 5-grams from social media text. To the best of our knowledge, there is no similar work that employs a topic-based approach using LSA for PI and SS tasks for social media text.
- We combine TLSA with sentiment analysis, which outperforms the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015: Paraphrase and Semantic Similarity in Twitter tasks.

2.2 Related Work

2.2.1 Corpus-Based algorithms

Corpus-based machine learning algorithms have an advantage over knowledge-based ones as they do not involve in human which can be expensive. The Google web 1T n-gram corpus [22] included all words appearing on the web in January, 2006 and is available in English and 10 European Languages [23]. This corpus has been used for text relatedness [90] and linguistic steganography [29]. The WaCky corpus of more than one billion words

[1] Reddit n-gram temporal corpus - `https://web.cs.dal.ca/~anh/?page_id=1699`

[2] Topic-based Latent Semantic Analysis - `http://cgm6.research.cs.dal.ca:8080/RedditFileDownload/tlsa.html`

from three languages, English, German, and Italian was introduced in 2009 by Baroni et al. [10]. It has been used in bilingual lexicography [62] and translators [128]. In 2010, Microsoft Web n-gram corpus provided all the word n-grams that are indexed by Bing search engine and provided through an XML web service [154]. Some notable usage includes textbox enriching [4] and social media language study [105]. Google Book n-gram corpus [116], introduced in 2012, includes all word n-grams found in Google book corpus from 1505 to 2008. Due to its yearly temporal characteristics, it has been used to study the changing psychology of culture [71], concepts of happiness [125], and mapping book to time [89]. Twitter n-gram corpus [80] only provides a small subset of social media n-grams in Twitter. As Twitter does not allow researchers to share full text of Tweets as a large corpus, it is not possible to collect, create and share terabyte-scale n-gram corpus for Tweets. Unlike Twitter, Reddit implements an open data policy and users can query any posted data on the website. Although OSNs have been studied intensively in recent years, there is no existing corpus that could be shared and provide insights from massive social network text. To the best of our knowledge, this new corpus is the first large-scale n-gram corpus that provides n-grams with a temporal feature (monthly) that is designed specifically for massive user-generated social media text.

2.2.2 Paraphrase Identification and Semantic Similarity

A summary of all the existing state-of-the-art paraphrase identification algorithms for traditional texts (e.g., newswire) using the Microsoft Research Paraphrase Corpus (MRPC) is in Par [127]. Although supervised approaches, such as typical machine learning classifiers using various feature sets [50, 93] and semantic text similarity [17, 110], achieved the best results, unsupervised methods using explicit semantic space [78], vector-based similarity [118], and WordNet similarity with matrix [61] also attained comparable results. With the increasing popularity of OSNs, researchers started to focus on the importance of developing paraphrase identification for social media text [162, 160, 73]. The results and findings support the hypothesis that informal language in social media with a high degree of lexical variations has posed serious challenges to both tasks. In this chapter, our focus is not the general PI or SS tasks but concentrates on the domain of social media.

The SemEval-2015 task 1 is the first competition that focuses on Paraphrase Identification and Semantic Similarity for social media text. There were 19 and 14 teams that participated in the PI and SS tasks, respectively. Most teams used supervised approach, for example, typical machine learning classifiers [58], neural networks [158], align and penalize architecture, semantic relatedness [150]. Two teams used unsupervised approaches (Orthogonal Matrix Factorization [74] and pre-trained word and phrase vectors on Google News dataset [158]) and one team uses semi-supervised approach that combines several word measures built from Rovereto Twitter n-gram corpus [80]. Our proposed approach will be compared and evaluated against these unsupervised and semi-supervised approaches.

Lately, large corpora are being used for the machine learning tasks. LSA has been widely used for paraphrase identification and semantic text analysis [78]. Guo and Diab [73] proposed Weighted Textual Matrix Factorization (WTMF), which is a novel latent model that captures the contextual meanings of words in sentences based on internal term-sentence matrix. This model uses both knowledge-based and large-scale corpus-based techniques to learn word representation. Our work uses the new corpus and introduces a novel approach to learn word representation that is dependent on the topic being discussed.

2.3 The New Reddit Temporal N-gram Corpus

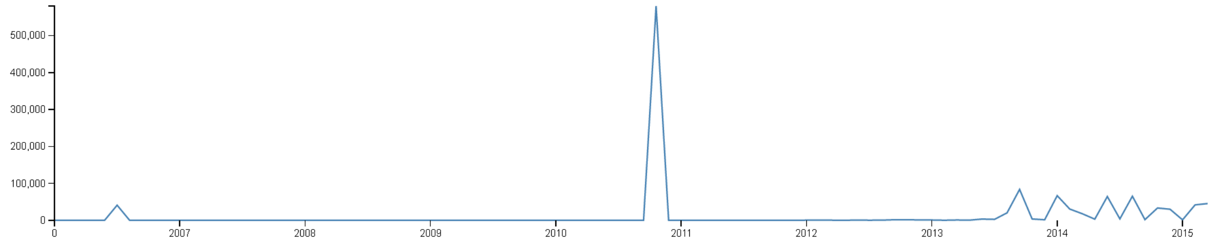
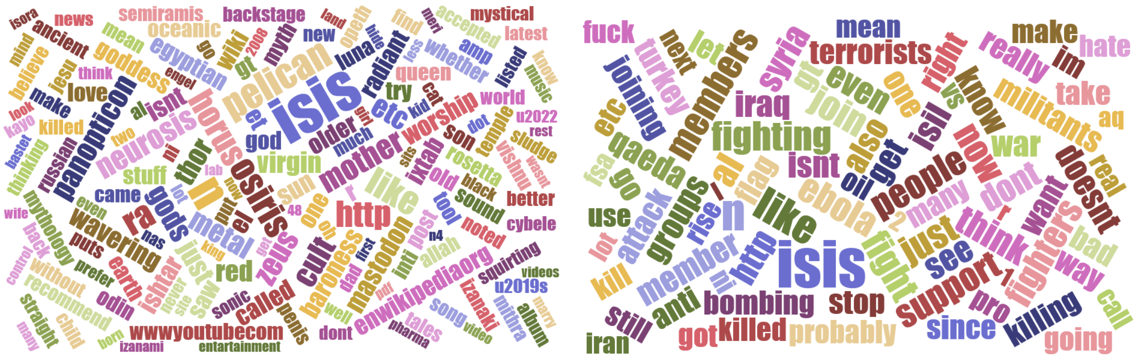


Figure 2.1: The frequency count of unigram "ISIS" in Reddit from 2007 to 2016. The x-axis represents the year while the y-axis shows the frequency count per month. The highest peak in the graph represents the rise of "ISIS" in October, 2010 following the outbreak of the Syrian Civil War in August, 2010.

We have created a word n-gram (1-5) corpus of 1.65 billion Reddit comments from October, 2007 until August, 2016 [12] using high performance distributed processing models on a cluster of 256 nodes with 16TB of shared memory. Most of the comments are in the English language. Each comment is separated into sentences, and each sentence is tokenized using Lucene (Apache). All the comments are lowercased. The Reddit comments are close to 2TB of text containing 135 billion sentences.

Each entry in the Reddit temporal n-gram corpus is an n-gram (1-5) and its frequency, month, and year from October, 2007 to August, 2016. The size of the corpus is 2.6 TB uncompressed. We show an example of each n-gram (1-5) in Table 2.1. This corpus can be accessed through downloadable files and a JSON web service which will return the frequency of an n-gram for each month. In addition, we implement a cloud-based visualization interface so that end users can query and analyze the social media n-grams in real time as shown in Figure 2.1. As our n-gram corpus is time-dependent, we also count the total number of occurrences of each n-gram for comparison with other corpora. A detailed statistical comparison between the new corpus and some other existing corpora is shown in Table 2.2. Although Microsoft Web n-gram is the largest n-gram corpus, they do not provide all the data to end users. Analyzing the whole Microsoft n-grams corpus is not practical through an XML web service. The Reddit temporal n-gram corpus is much larger than the Google Web 1T n-gram corpus. One of the reasons is that Google Web 1T n-gram corpus only keeps unigrams with more than 200 frequency counts and other n-grams with more than 40 frequency counts. After analyzing the Google Web corpus, we found that although our corpus has a larger vocabulary than the Google one, the frequency count for each n-gram is lower. This confirms the noise and shortness hypothesis of social media text. We decide to keep all the raw n-grams of the new corpus to preserve these characteristics of social media text. We illustrate in Figure 2.2 how the Reddit temporal n-gram corpus shows the evolution of the word "ISIS".



(a)

(b)

Figure 2.2: An example of word cloud showing the context words of "ISIS" in the 5-grams of the corpus before and after August, 2010. a) "ISIS" is mainly discussed as an Egyptian god before August, 2010, b) "ISIS" means the Islamic State in Iraq and Syria after August, 2010.

2.4 Topic-based Latent Semantic Analysis

We first formulate the approach for TLSA. Consider a list of topics $T = \{T_1, T_2, ..., T_l\}$ and each topic T_i has a list of pairs of Tweets $P = \{(t_{11}, t_{12}), (t_{21}, t_{22}), ..., (t_{m1}, t_{m2})\}$ where

Table 2.1: Examples of n-grams (1-5) of the newly created n-gram corpus about the current US president "Donald Trump". Each entry includes the word (n-gram), its frequency, its month, and its year from October 2007 to August 2016.

	word	frequency	year	month
1-gram	trump	981	2015	01
2-gram	trump apprentice	31	2015	01
3-gram	donald trump battle	16	2015	01
4-gram	donald trump ignorant tweet	8	2015	01
5-gram	take donald trump advice in	2	2015	01

Table 2.2: Statistical comparison between Reddit temporal n-gram corpus and its counterparts.

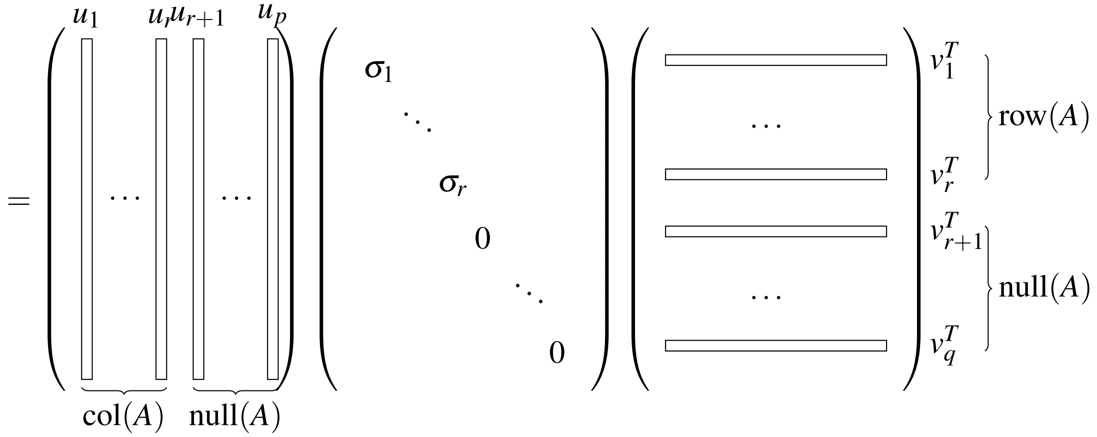
Corpus	1-gram	2-gram	3-gram	4-gram	5-gram
Google web 1T n-gram corpus	13.5M	314M	977M	1.3B	1.12B
Microsoft web n-gram corpus	1.2B	11.7B	60.1B	148.5B	237B
Reddit temporal n-gram corpus	170.2M	1.2B	6.7B	18.4B	30.1B

each pair is evaluated for PI and SS tasks. For each topic T_i, we construct a list of unigrams $O = \{o_1, o_2, ..., o_p\}$ from P and a list of 5-grams $F = \{f_1, f_2, ..., f_q\}$ from Reddit temporal n-gram corpus where f_i contains the topic T_i. Next, we construct the unigram/5-gram matrix X from O and F.

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1q} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2q} \\ \dots & \dots & \dots & \dots & \dots \\ x_{p1} & x_{p2} & x_{p3} & \dots & x_{pq} \end{bmatrix}$$

where each row r_i represents the occurrence of a unigram term u_i to all 5-grams in F and x_{ij} describes the occurrence of unigram o_i in a 5-gram f_j plus the frequency of the 5-gram f_j in the Reddit temporal n-gram corpus. This matrix considers both the relation between a word with other words in a 5-gram and with the frequency of this 5-gram in the corpus. Next, we decompose matrix X using a Singular Value Decomposition (SVD):

$X = U\Sigma V^T$



where Σ is a diagonal matrix that contains the singular values in descending values. U and V are orthogonal matrices that contain the left and right singular vectors respectively.

Next, for each sentence s_i in topic T_i, we construct a vector $\vec{v}$ which represents the occurrence of s_i in the list of unigrams of topic T_i. This vector is translated into a sentence vector representation by the following formula:

$$\vec{v} = \vec{v} * U_k * S_k$$

where k is the chosen k singular values which show the dimensions with the greatest variance between words and documents (the value of k is explained in Section 2.6.3). Finally, the semantic similarity between two sentences is calculated using the cosine similarity between their vectors.

Due to the enormous size of the Reddit temporal n-gram corpus, selecting the related 5-grams for each topic is not feasible using a traditional relational database system. We tried to load our data into IBM Netezza data warehouse but the query time was not reasonable for a real-time system. We load all the corpus data to Google Bigquery. For each topic T_i, we query all the related 5-grams f_i using Google Bigquery regular expression "word like (%T_i%)" where % represents the wild card search. After constructing matrix X, we use Microsoft Azure Apache Spark for SVD decomposition. A summary of the proposed approach is shown in Figure 2.3.

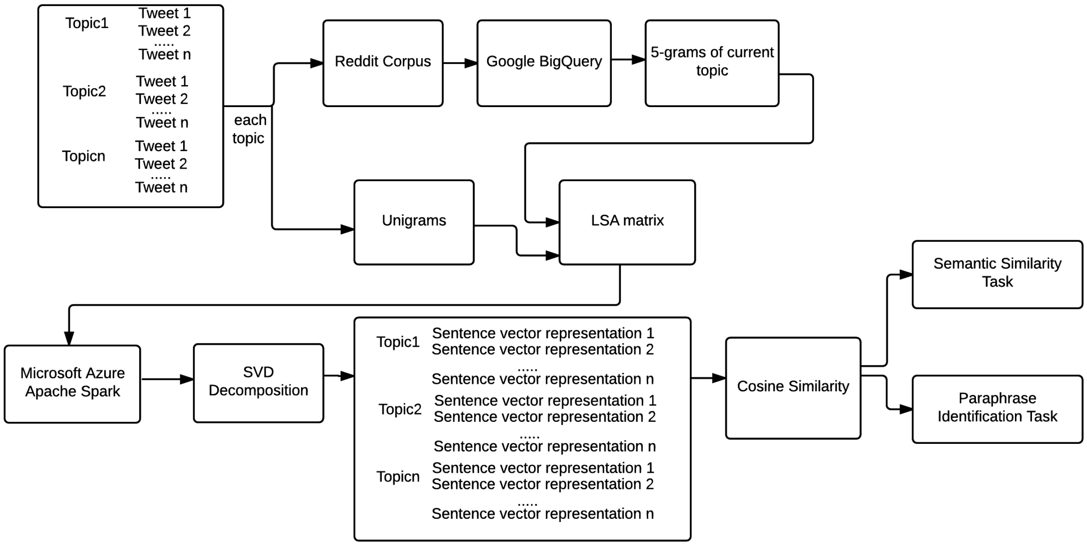

Topic1
Tweet 1
Tweet 2
.....
Tweet n
Topic2
Tweet 1
Tweet 2
.....
Tweet n
Topicn
Tweet 1
Tweet 2
.....
Tweet n
each topic
Reddit Corpus
Google BigQuery
5-grams of current topic
Unigrams
LSA matrix
Microsoft Azure Apache Spark
SVD Decomposition
Topic1
Sentence vector representation 1
Sentence vector representation 2
.....
Sentence vector representation n
Topic2
Sentence vector representation 1
Sentence vector representation 2
.....
Sentence vector representation n
Topicn
Sentence vector representation 1
Sentence vector representation 2
.....
Sentence vector representation n
Cosine Similarity
Semantic Similarity Task
Paraphrase Identification Task

Figure 2.3: The proposed Topic-based Latent Semantic Analysis using distributed parallel computing, Google BigQuery, and Microsoft Azure Apache Spark. The list of topics and pairs of Tweets are from the test dataset of SEMEVAL 2015 Task 1. The semantic similarity between two sentences is computed with regard to a specific topic being discussed in two sentences.

2.5 Evaluations of Paraphrase Identification and Semantic Similarity for Social Media Text

To evaluate the performance of TLSA algorithm, we use the PIT-2015 Twitter dataset [159]. Although this approach uses PIT-2015 dataset for evaluation, it can be extended to any general topic-based datasets. The PIT-2015 dataset includes 17,790 sentence pairs for training and 972 test sentence pairs which were annotated and developed by Xu et al. [159]. The dataset was constructed from Twitter data and has intrinsic characteristics from social network data: (i) opinionated and colloquial sentences from realistic social media text; (ii) lexically diverse pairs of sentences for paraphrases; and (iii) sentences that seem lexically similar but semantically dissimilar [158]. Example pairs of sentences for paraphrase, non-paraphrase, and debatable cases are shown in Table 2.3. The detailed statistics of this ground-truth dataset is shown in Table 2.4. Each sentence is processed with tokenization, part-of-speech and named entity tags and each sentence pair is annotated by experts. In the test set, there are 972 sentence pairs collected from Twitter in 20 trending topics between

Table 2.3: Examples of Paraphrase Identification and Semantic Similarity sentence pairs. All three sentence pairs are about the movie “8 Mile” which is a topic for TLSA. A sentence pair is a paraphrase if its Pearson Correlation score is above 0.6. A sentence pair is a non-paraphrase if its Pearson Correlation score is below 0.6. A sentence pair is debatable if its Pearson Correlation score is equal to 0.6.

Topic	Paraphase	Sentence 1	Sentence 2
8 mile	True	The Ending to 8 Mile is my fav part of the whole movie	Those last 3 battles in 8 Mile are THE shit
8 mile	False	All the home alones watching 8 mile	The last rap battle in 8 Mile nevr gets old ahah
8 mile	Debatable	8 mile is just a classic	After watching 8 mile I feel like such a thug

May 13th and June 10th, 2013. As mentioned in Das and Smith [50], some algorithms may work well specifically for MRPC because of its imbalanced nature (lack of non-paraphrases). PIT-2015 Twitter dataset is more balanced as it contains 70% non-paraphrases and the 34% paraphrases.

Table 2.4: PIT-2015 Twitter dataset. The test data is more balanced than MRPC as it has a higher percentage of non-paraphrase sentence pairs. The unsupervised TLSA only uses the test data for evaluation.

	Sent Pairs	Paraphase	Non-paraphrase	Debatable
Train	13063	3996 (30.6%)	7534 (57.7%)	1533 (11.7%)
Test	972	175 (18.0%)	663 (68.2%)	134 (13.8%)

2.5.1 Task 1 — Paraphrase Identification and Evaluation Metrics

For a specific topic, given two sentences, the system has to determine if two sentences have the same or similar meaning and discuss the same topic. For two non-paraphrase sentence pairs, the sentence pair discussing the same topic has a higher score than the sentence pair discussing an unrelated topic. Precision, recall, and F1 (harmonic mean of precision and recall) are used as evaluation metrics.

2.5.2 Task 2 — Semantic Similarity and Evaluation Metrics

For a specific topic, given two sentences, the system has to give a score between 0 (no relation) and 1 (semantic equivalence) to represent their semantic equivalence. For two sentence pairs, the sentence pair discussing the same topic has a higher semantic similarity score than the sentence pair discussing an unrelated topic. Pearson correlation is used as an evaluation metric.

2.6 Evaluation

2.6.1 Baselines

We used first two baselines from [158] and introduced two new baselines that are more related to the proposed corpus-based and topic-based LSA.

Random: Each sentence pair is assigned a random real semantic similarity score between [0, 1]. For PI task, this baseline applies 0.5 as a cutoff (paraphrase if semantic similarity score is above 0.5).

Weighted Matrix Factorization (WTMF): This baseline uses the state-of-the-art unsupervised method of Guo and Diab [73]. It not only considers the semantic space of words present in the data but also missing words from the sentences. This feature is designed specifically for short texts in social media. Finally, the value 0.5 is used as a cutoff for the PI task.

Random 5-gram: This baseline determines whether introducing the use of topics in LSA improves the accuracy of both PI and SS tasks for SEMEVAL 2015. To construct matrix X, we select random 5-grams from the Reddit temporal n-gram corpus with the same size of the 5-grams that contain the topic.

Google Tri-gram Method (GTM): Google Tri-gram Method [90] assigns a semantic similarity score between two sentences using the unigrams and trigrams of the Google Web 1T corpus. We also use 0.5 as a cutoff for the PI task.

2.6.2 SEMEVAL 2015 Unsupervised and Semi-supervised Methods

Columbia: This method used Orthogonal Matrix Factorization to compute a representation vector for each sentence [74] and then computes a similarity score based on these vectors [158].

Yamraj: This method learned sentence vectors from Google News dataset (about 100 billion words) and Wikipedia articles. Cosine distance is used to compute the vector similarity scores.

MathLingBp: This method exploits the use of the align-and-penalize architecture of Han et al. [77] and adopts the use of several word similarity metrics using a semi-supervised approach [158].

2.6.3 Experimental Results

First, we compare the performance of TLSA with various parameters, such as the number of singular values and the dimensionality of the 5-grams. For SS task, we achieved the best result for SS task when the singular value *k* is equal to 80 with an increasing 5-gram dimensionality size as shown in Figure 2.4. In addition, for SS task, the Pearson correlation is not improving when the number of 5-grams is above 1M.

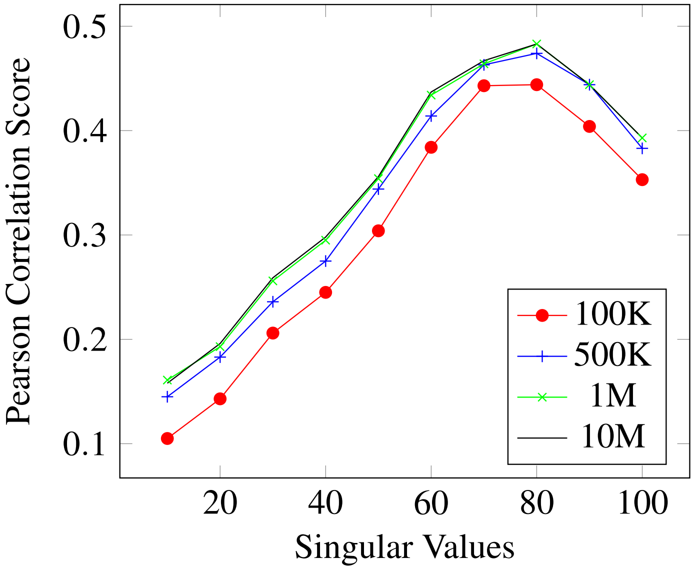


Figure 2.4: For the Semantic Similarity task, Pearson correlation score with an increasing singular values and 5-gram dimensionality. TLSA achieves the best Pearson correlation score for *k* = 80 and the dimensionality of 5-grams = 1M.

2.6.3.1 Topic-based LSA versus Baselines and other Methods

This section compares the proposed approach with the baselines and SEMEVAL 2015 unsupervised and semi-supervised methods. As shown in Table 2.5, TLSA achieved the best result for the SS task (Pearson correlation) compared with all the baselines and compared methods. This means that training an LSA model using topic-based 5-gram helps increase the result of PI and SS tasks. For the PI task, observing that the semantic similarity scores for sentence pairs are either very high or very low, we tried two cutoffs 0.25 and 0.5 (SEMEVAL 2015 allows two runs per team) and TLSA outperforms all the baselines. With a low cutoff value, TLSA achieves a high precision and a low recall. To improve the PI results, we assumed that two sentences are paraphrases only if they have the same sentiment scores (e.g., both are positives or negatives). Based on this assumption, each sentence is assigned a sentiment score using OpenNLP. Adding sentiment analysis to TLSA (i.e., TLSA & Sentiment) outperforms all the baselines and compared methods. Another important observation is that although our unsupervised approach achieves the best results against the baselines and compared methods, its results are still not comparable to human upper-bound from the dataset and other supervised approaches. This means that improving the results of PI and SS tasks for social media text using an unsupervised approach is still a challenge for researchers.

Table 2.5: TLSA results with other baselines and compared methods. Combining TLSA with sentiment analysis achieves the best result for both PI and SS tasks.

Methods / *Baselines*	Paraphrase Identification			Semantic Similarity			
	F1	Precision	Recall	Pearson	maxF1	maxPrec	maxRecall
Human Upperbound	0.823	0.752	0.0908	0.735	–	–	–
TLSA & Sentiment	**0.591**	0.764	0.480	**0.483**	0.582	0.761	0.472
COLUMBIA	0.588	0.593	0.583	0.425	0.599	0.623	0.577
TLSA	**0.585**	0.761	0.474	**0.483**	0.585	0.761	0.474
YAMRAJ	0.496	0.725	0.377	0.360	0.542	0.502	0.589
WTMF	0.536	0.450	0.663	0.350	0.587	0.570	0.606
Random 5-gram	0.504	0.716	0.389	0.466	0.564	0.824	0.429
GTM	0.495	0.391	0.674	0.371	0.582	0.761	0.472
Random	0.266	0.192	0.434	0.017	0.350	0.215	0.949

2.7 Conclusions

In this chapter, we introduced Reddit temporal n-gram corpus, which is designed specifically for social media text. We create the corpus using distributed parallel computing and implement a cloud-based visualization interface so that end users can query any n-grams from the corpus. Both the corpus and the interface are publicly available in this URL - Reddit n-gram temporal corpus. This large-scale terabyte corpus includes all the word unigram to 5-gram, and their frequency per month from October, 2007 to August, 2016.

To show the usefulness of this corpus, we propose a novel Topic-based Latent Semantic Analysis approach which exploits the 5-grams of the corpus. The proposed TLSA outperforms all the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015 Task 1 — Semantic Similarity for the PIT-2015 dataset. Combining with sentiment analysis, the proposed approach also achieves the best result for the Paraphrase Identification of SEMEVAL 2015 Task 1. In addition, TLSA is language-independent and scalable for the large-scale nature of social media text.

For future work, we aim to use this corpus to study the linguistic patterns of social media text and finding the meaning of new words in social media. We also plan to integrate this proposed semantic similarity score into our existing work to improve the results of meme clustering tasks [42] and rumour detection and visualization framework [45].

Chapter 3

An Offline-Online Visual Framework for Clustering Memes in Social Media

The amount of data generated in Online Social Networks (OSNs) is increasing every day. Extracting and understanding trending topics and events from the vast amount of data is an important area of research in OSNs. This chapter proposes a novel clustering framework to detect the spread of memes in OSNs in real time. The offline-online meme clustering framework exploits various similarity scores between different elements of Reddit submissions, two strategies to combine those scores based on Wikipedia concepts as an external knowledge, text semantic similarity and a modified version of Jaccard Coefficient. The two combination strategies include: (1) automatically computing the similarity score weighting factors for five elements of a submission and (2) allowing users to engage in the clustering process and filter out outlier submissions, modify submission class labels, or assign different similarity score weight factors for various elements of a submission using a visualization prototype. The offline-online clustering process does a one-pass clustering for existing OSN data in the first step by calculating and summarizing each cluster statistics using Wikipedia concepts. For the online component, it assigns new streaming data points to the appropriate clusters using a modified version of online k-means. The experiment results show that the use of Wikipedia as external knowledge and text semantic similarity improves the speed and accuracy of the meme clustering problem when comparing to baselines. For the online clustering process, using a damped window model approach is suitable for online streaming environments as it not only requires low prediction and training costs, but also assigns more weight to recent data and popular topics. The basis of this chapter is from the published papers [41, 43].

3.1 Introduction

Online Social Networks (OSNs) are networks of online interactions and relationships that are formed and maintained through various social networking sites such as Facebook, LinkedIn, Reddit, and Twitter. Nowadays, hundreds of millions of people and organizations turn to OSNs to interact with one another, share information, and connect with friends and strangers. OSNs have been especially useful for disseminating information in the context of political campaigning, news reporting, marketing, and entertainment [95].

OSNs have been recently used as an effective source for end users to know about breaking-news or emerging memes. A meme is a unit of information that can be passed from person to person in OSNs [87]. Despite their usefulness and popularity, OSNs also have a “negative” side. As well as spreading credible information, OSNs can also spread rumours, which are truth-unverifiable statements. For example, so many rumour-driven memes about swine flu outbreak (e.g., “swine flu pandemic meme” in Figure 3.1) were communicated via OSNs in 2009 that the US government had to tackle it officially on their website [119, 60]. Problems like these (i.e., rumour-driven memes going viral) are unfortunately not isolated and prompt the question of how to identify and limit the spread of rumours in OSNs. In order to detect rumours, we have to identify memes that are rumour-related in OSNs. Clustering is a simple and efficient unsupervised process to identify memes in OSNs by grouping similar information into the same category. However, traditional clustering algorithms do not work effectively in OSNs due to the heterogeneous nature of social network data [91]. Labelling massive amounts of social network data is an intensive task for classification. To overcome these limitations, this chapter proposes a semi-supervised approach with relevance user feedback for detecting the spread of memes in OSNs.

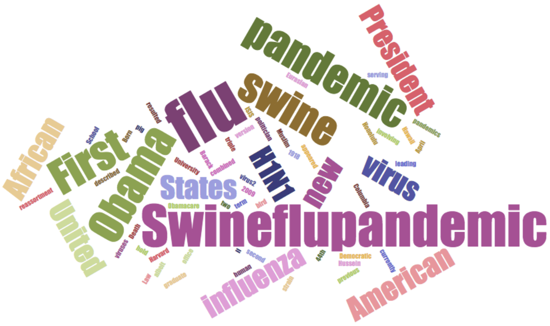
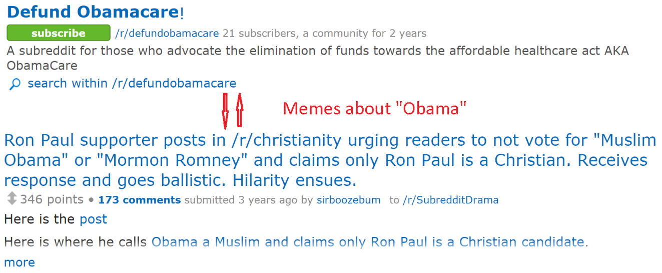


Figure 3.1: A word cloud example of popular memes in OSNs.

In text clustering, a similarity measure is a function that assigns a score to a pair of texts in a corpus that shows how similar the two texts are. Computing similarity scores between texts is one of the most computationally intensive and important steps for producing a good clustering result [15]. For a meme clustering task, this process is usually hindered by the lack of significant amounts of textual content, which is an intrinsic characteristic of OSNs [91, 85]. For example, in Reddit.com, most submission titles are very short and concise. Although the title of a submission may provide meaningful information about the topic, the titles may not provide enough information to determine if two submissions are discussing the same topic. In Figure 3.2, two Reddit submissions are both talking about "Obama", but one is discussing the meme "Obamacare", while the other is discussing the rumour-related meme "Obama is a Muslim". The sparsity of Reddit submission title texts significantly contributes to the poor performance of traditional text clustering techniques for grouping submissions into the same category. We, therefore, propose strategies to leverage the use of references to external content.



Defund Obamacare!

/r/defundobamacare 21 subscribers, a community for 2 years

A subreddit for those who advocate the elimination of funds towards the affordable healthcare act AKA ObamaCare

search within /r/defundobamacare

Ron Paul supporter posts in /r/christianity urging readers to not vote for "Muslim Obama" or "Mormon Romney" and claims only Ron Paul is a Christian. Receives response and goes ballistic. Hilarity ensues.

346 points • 173 comments submitted 3 years ago by sirboozebum to /r/SubredditDrama

Here is the post

Here is where he calls Obama a Muslim and claims only Ron Paul is a Christian candidate.

more

Figure 3.2: Reddit submissions about the same meme "Obama". The top submission discusses the meme "Obamacare", while the bottom submission discusses the meme "Obama is a Muslim"

A submission may include one or more comments from users, which discuss the submission topic. It can also contain a URL that points to an external article that further discusses the topic of the submission. Similarly, a submission may include an image that also provides more valuable information about the submission topic. By introducing the use of comments, URL content, and image content of a submission, we exploit more valuable data for text clustering tasks, which helps detect memes in OSNs more efficiently.

Vector space models are commonly used to represent texts for a clustering task. In these models, each text is represented as a vector where each element corresponds to an attribute extracted from the text. One of the benefits of these models is their simplicity in calculating the similarity between two vectors based on linear algebra. The two most famous models are

TF-IDF (Term Frequency — Inverse Term Frequency) and Bag-of-Words. However, those models rely solely on lexical representation of the texts, which does not capture semantic relatedness between words in a text. For example, the use of polysemy and synonymy are very popular in several types of texts and play an important role in determining whether two words, concepts or texts are semantically similar. This motivates many researchers to explore the advantage of semantic similarity in the task of text clustering by utilizing word relatedness through external thesauruses like WordNet and Wikipedia [129, 88]. However, they remain far from covering every word and concept used in OSNs. This chapter explores Google n-grams algorithm of Islam et al. [90] which uses Google n-grams dataset to compute the relatedness between words for computing similarity scores, and proposes two novel strategies to combine those scores for the task of clustering memes.

Although clustering streaming and time series data are established fields in the area of text clustering [148, 32, 75, 68, 66], clustering memes in OSNs has just started to gain attention recently [166, 3, 1, 91]. OSN data has both characteristics of streaming and time series data, as well as another important characteristic. The volume of OSN data is massive and can not be handled efficiently by traditional streaming and time series clustering algorithms [91]. In order to tackle that problem, we propose a novel approach to speed up the processing of online meme clustering that uses both semantic similarity and Wikipedia concepts to efficiently store and summarize OSN data in real time.

With the increasing amount of online social network data, understanding and analyzing them is becoming more challenging. Researchers have started to employ human's ability to effectively gain visual insight on data analysis tasks. The task of clustering memes shares some similarity with clustering text, but they are also intrinsically different. For example, social network data is usually poorly-written and content-limited. This reduces the quality of clustering results. For a Reddit submission, the relationships between the title, comment, image, and URL sometimes are disconnected (e.g., a title has a different subject from the content). In this chapter, we developed a visualization prototype to allow users to better distinguish the similarity between submissions and use this feedback to improve the clustering results.

This chapter extends previous work of Dang et al. [41] by formalizing the problem of meme clustering and proposes a novel approach for clustering Reddit submissions. It makes the following contributions:

- Extends and improves the similarity scores between different elements of Reddit submission of Dang et al. [41] by introducing the use of Wikipedia concepts as an external knowledge.
- Introduces a modified version of Jaccard Coefficient that employs the use of text semantic similarity when comparing the similarity score between two sets of Wikipedia concepts.
- Proposes an offline-online clustering algorithm that exploits semantic similarity and Wikipedia concepts to achieve good clustering results in real time. The offline clustering component computes and summarizes cluster statistics to speed up the process of the online clustering component. In addition, for each cluster, we adopt the damped window model and propose a novel approach to summarize each cluster as a set of Wikipedia concepts where each concept is assigned a weight based on its recency and popularity. The online clustering component applies a semantic version of Jaccard Coefficient.
- The experiments show the use of Wikipedia concepts increases the accuracy result of the meme clustering tasks. Although only using Wikipedia concepts as a similarity score does not increase the clustering result, using both Wikipedia concepts and text semantic similarity increase the clustering accuracy for both offline and online clustering components.

3.2 Related Work

This section presents current research on text semantic similarity and detecting the spread of memes in OSNs.

3.2.1 Similarity Measures and Text Clustering

Several similarity measures have been proposed in the literature for the task of text clustering. The most popular ones are lexical measures like Euclidean, Cosine, Pearson Correlation, and Extended Jaccard measures. Strehl et al. [146] provided a comprehensive study on using different clustering algorithms with these people measures. The authors used several clustering algorithms on the YAHOO dataset, and showed that Extended Jaccard and Cosine similarity performed better and achieved results that are close to a human-labelling process. However, lexical similarity measures do not consider the semantic similarity between words in the texts.

Some researchers have taken advantage of the semantic relatedness of texts by using external resources to enrich word representation. In Pedersen et al. [129], the authors suggested using WordNet as a knowledge base to determine the semantic similarity between words. The experiment results have shown that external knowledge bases like WordNet improve the clustering results in comparison to the Bag-of-Words models. Hu et al. [88] proposed the use of Wikipedia as external knowledge for text clustering. The authors tried to match concepts in text into Wikipedia concepts and categories. Similarity scores between concepts are calculated based on the text content information, as well as Wikipedia concepts, and categories. The experiment results have shown that using Wikipedia as external knowledge provided a better result than using WordNet due to the limited coverage of WordNet. Bollegala et al. [19] proposed the use of information available on the Web to compute text semantic similarity by exploiting page counts and text snippets returned by a search engine. Our work is intuitively different from these approaches, as it introduces the use of word relatedness based on the Google n-grams dataset [21]. The proposed semantic similarity scores between texts are calculated based on that algorithm to handle the low quality (i.e. poor writing) of social network data. Using Google n-grams dataset as external knowledge is more effective than textual as well as other semantic approaches, as the Google n-grams dataset has more coverage than other semantic approaches.

3.2.2 Online Clustering Algorithms in OSNs

This section discusses the related work of event detection and online meme clustering in OSNs. The proposed online meme clustering algorithm takes advantage of the current work of both clustering streaming data and clustering time series data. Clustering streaming data has been actively researched in the literature. Aggarwal et al. [3] proposed a graph-based sketch structure to maintain a large number of edges and nodes at the cost of potential loss of accuracy. Zhao and Yu [166] extended the graph-based clustering streaming algorithms with side information, such as user metadata in OSNs. As OSN data is dependent on its temporal context, time series is another important feature of clustering streaming data algorithms. We present related work of the three types of time series clustering algorithms in the literature: (1) landmark window approach, (2) sliding window approach, and (3) damped window approach [131]. In clustering streaming data, landmark-based models consider all the historical data from a landmark time and all data have an equal weight [112, 27, 68]. Sliding-based models are common stream processing models which only examine data at a fixed-time window (e.g., last 5 minutes or last 24 hours) [75, 32, 103]. Damped window models introduced the use of decay variable to replace old data to increase the accuracy of streaming clustering results [66, 30]. JafariAsbagh et al. [91] used a sliding window approach for detecting memes in real time that does not consider the topic evolution and persistence. As the spread of memes in OSNs is dependent on the meme topics and its context [82], the proposed online meme clustering algorithm explores the damped window approach which consider the frequency and recency of memes. Researchers also investigate if the use of external knowledge (e.g., Wikipedia) helps the clustering results for social media texts. Banerjee et al. [9] introduced the use of Wikipedia as external knowledge to improve the accuracy results for short texts. Dang et al. [41] used text semantic similarity computed from Google n-grams dataset to alleviate the problem of shortness and noise of OSN data.

Scientists also explored the use of visualization for text clustering with relevance user feedback. Lee et al. [98] introduced iVisClustering, an interactive visualization framework based on LDA topic modelling. This system provides some interactive features, such as removing documents or clusters, moving a document from one cluster to another, merging two clusters, and influencing term weights. Choo et al. [34] presented an interactive visualization for dimension reduction and clustering for large-scale high-dimensional data. The system allows users to interactively try different dimension reduction techniques and clustering algorithms to optimize the clustering results. One of the limitations of these systems is that they focus on the clustering algorithms and results and have limited supports for combining similarity scores for different parts of a text (e.g., the title and body of a text). This chapter introduces a visualization prototype to combine different similarity scores for our clustering process interactively and incrementally.

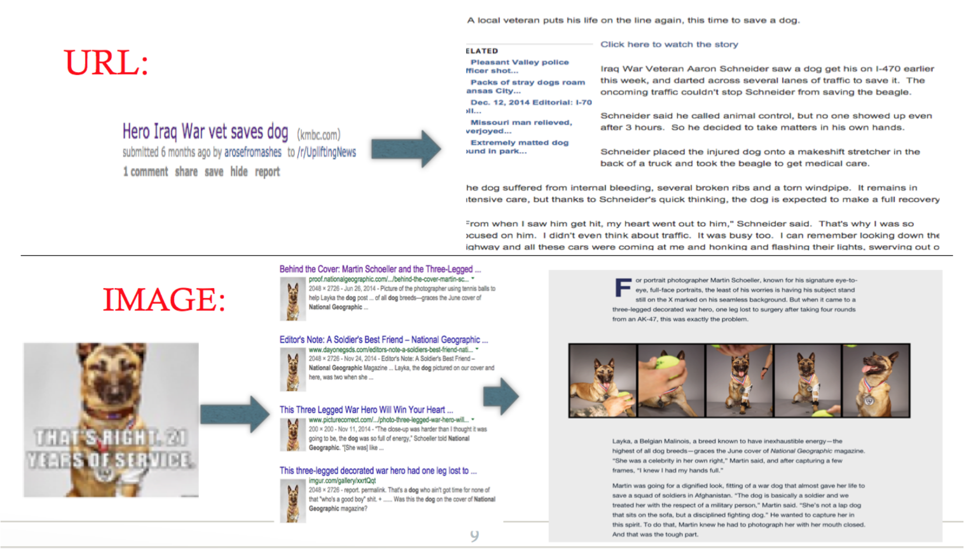
3.2.3 Detecting Memes in Online Social Networks

Recently, researchers have started adapting state of the art clustering algorithms to OSN data. Leskovec et al. [100] proposed a meme-tracking framework to monitor memes that travel through the Web in real-time. The framework studied the signature path and topic of each meme by grouping similar short, distinctive phrases together. One drawback of this framework is that it only applies lexical content similarity to detect memes. This did not work well for memes that are related but not using the same words, and those that are short and concise (e.g., Tweets on Twitter). Cataldi et al. [26] proposed an approach that monitored the real-time spread of emerging memes in Twitter. The authors defined an emerging term as one whose frequency of appearance had risen within a short period and had not emerged or was only rarely discussed in the past. A navigable topic graph is constructed to connect semantically related emerging terms. Emerging memes are extracted from this graph based on semantic relationships between terms over a specified time interval. Becker et al. [13] formulated the problem of clustering for event detection and proposed a supervised approach to classify tweets using a predefined set of features. The proposed approach includes various types of features: textual, temporal, and spatial. Aggarwal and Subbian [2] presented a clustering algorithm that exploits both content and network-based features to detect events in social streams. The proposed algorithm uses knowledge about metadata of Twitter users. Thom et al. [148] developed a system for interactive analysis of location-based microblog messages, which can assist in the detection of real-world events in real time. This approach uses X-means, a modified version of K-means, to detect emerging events. Finally, JafariAsbagh et al. [91] introduced an online meme clustering framework using the concept of Protomemes. Each Protomeme is defined based on one of the atomic information entities in Twitter: hashtags, mentions, URLs, and tweet content. An example of Protomeme is the set of tweets containing the hashtag #All4Given. This approach uses a sliding window model that can lead to good offline prediction accuracy but not suitable for online streaming environments. As online meme clustering algorithms require low prediction and training costs, our proposed online meme clustering algorithm stores cluster summary statistics using Wikipedia concepts and applies a damped window approach with offline-online components for clustering memes in OSNs. Although Twitter has been the most popular OSN for detecting memes, little work has been done to detect rumour-related memes on Reddit.

3.3 Reddit Social Network

Reddit, which claims to be “the front page of the internet”, is a social news website, where users, called redditors, can create a submission or post direct links to other online content. Other redditors can comment or vote to decide the rank of this submission on the site. Reddit has many subcategories, called sub-reddits that are organized by areas of interests. The site has a large base of users who discuss a wide range of topics daily, such as politics and world events. Alexa ranks Reddit.com as the 24th most visited site globally. Each Reddit submission has the following elements:

- **Title:** The title summarizes the topic of that submission. The title text is usually very



Hero Iraq War vet saves dog (kmbc.com)
submitted 6 months ago by arosefromashes to /r/UpliftingNews
1 comment share save hide report

A local veteran puts his life on the line again, this time to save a dog.

ELATED
Pleasant Valley police fficer shot...
Packs of stray dogs roam ansas City...
Dec. 12, 2014 Editorial: I-70 oll...
Missouri man relieved, verjoyed...
Extremely matted dog ound in park...

Click here to watch the story

Iraq War Veteran Aaron Schneider saw a dog get his on I-470 earlier this week, and darted across several lanes of traffic to save it. The oncoming traffic couldn't stop Schneider from saving the beagle.

Schneider said he called animal control, but no one showed up even after 3 hours. So he decided to take matters in his own hands.

Schneider placed the injured dog onto a makeshift stretcher in the back of a truck and took the beagle to get medical care.

he dog suffered from internal bleeding, several broken ribs and a torn windpipe. It remains in itensive care, but thanks to Schneider's quick thinking, the dog is expected to make a full recovery

From when I saw him get hit, my heart went out to him," Schneider said. That's why I was so ocused on him. I didn't even think about traffic. It was busy too. I can remember looking down the ighway and all these cars were coming at me and honking and flashing their lights, swerving out o

Behind the Cover: Martin Schoeller and the Three-Legged ...
proof.nationalgeographic.com/.../behind-the-cover-martin-sc...
2048 × 2726 - Jun 26, 2014 - Picture of the photographer using tennis balls to help Layka the **dog** post ... of all **dog** breeds—graces the June cover of **National Geographic** ...

Editor's Note: A Soldier's Best Friend – National Geographic ...
www.dayonegsds.com/editors-note-a-soldiers-best-friend-nati...
2048 × 2726 - Nov 24, 2014 - Editor's Note: A Soldier's Best Friend – **National Geographic** Magazine ... Layka, the **dog** pictured on our cover and here, was two when she ...

This Three Legged War Hero Will Win Your Heart ...
www.picturecorrect.com/.../photo-three-legged-war-hero-will...
200 × 200 - Nov 11, 2014 - "The close-up was harder than I thought it was going to be, the **dog** was so full of energy," Schoeller told **National Geographic.** "[She was] like ...

This three-legged decorated war hero had one leg lost to ...
imgur.com/gallery/xxrtQqt
2048 × 2726 - report. permalink. That's a **dog** who ain't got time for none of that "who's a good boy" shit. + ... Was this the **dog** on the cover of **National Geographic** magazine?

For portrait photographer Martin Schoeller, known for his signature eye-to-eye, full-face portraits, the least of his worries is having his subject stand still on the X marked on his seamless background. But when it came to a three-legged decorated war hero, one leg lost to surgery after taking four rounds from an AK-47, this was exactly the problem.

Layka, a Belgian Malinois, a breed known to have inexhaustible energy—the highest of all dog breeds—graces the June cover of *National Geographic* magazine. "She was a celebrity in her own right," Martin said, and after capturing a few frames, "I knew I had my hands full."

Martin was going for a dignified look, fitting of a war dog that almost gave her life to save a squad of soldiers in Afghanistan. "The dog is basically a soldier and we treated her with the respect of a military person," Martin said. "She's not a lap dog that sits on the sofa, but a disciplined fighting dog." He wanted to capture her in this spirit. To do that, Martin knew he had to photograph her with her mouth closed. And that was the tough part.

Figure 3.3: External content from image and URL. The top submission has an URL and we extracted the URL content. The bottom submission has an image and we extracted the text of the image from Google Reverse Image Search.

$$
\mathrm{GTM}(\omega_1,\omega_2)=\begin{cases}
\dfrac{\log\frac{\mu_T(\omega_1,\omega_2)C_{\max}^2}{C(\omega_1)C(\omega_2)\min(C(\omega_1)C(\omega_2))}}{-2\times\log\frac{\min(C(\omega_1),C(\omega_2))}{C_{\max}}} & \text{if } \log\dfrac{\mu_T(\omega_1,\omega_2)C_{\max}^2}{C(\omega_1)C(\omega_2)\min(C(\omega_1)C(\omega_2))}>1\\[2ex]
\dfrac{\log 1.01}{-2\times\log\frac{\min(C(\omega_1),C(\omega_2))}{C_{\max}}} & \text{if } \log\dfrac{\mu_T(\omega_1,\omega_2)C_{\max}^2}{C(\omega_1)C(\omega_2)\min(C(\omega_1)C(\omega_2))}\leq 1\\[2ex]
0 & \text{if } \mu_T(\omega_1,\omega_2)=0
\end{cases}
$$

Figure 3.4: GTM semantic similarity calculation.

short and concise. The title may also have a description to further explain it.

- **Comments:** Users can post a comment that expresses their opinions about the corresponding submission or other user comments. Users can also vote comments up or down.
- **URL:** Each submission may contain a link to an external source of information (e.g., news articles) that is related to the submission.
- **Image:** Submissions may also have a link to an image that illustrates the topic of the submission.

Figure 3.3 explains how to collect image and URL content from Reddit submissions. Unlike other OSNs, Reddit is fundamentally different in that it implements an open data policy; users can query any posted data on the website. For example, other OSNs, like Twitter or Facebook, allows circulating information through a known cycle (e.g., "friend" connections), whereas Reddit promotes a stream of links to all users in a simple bookmarking interface. This makes Reddit a more effective resource to study the spread of memes in OSNs. To the best of our knowledge, no similar work has been done on clustering memes in Reddit.

3.4 Google Tri-gram Method

Google Tri-gram Method (GTM) [90] is an unsupervised corpus-based approach for computing semantic relatedness between texts. GTM uses the uni-grams and tri-grams of the Google Web 1T N-grams corpus [90] to calculate the relatedness between words, and then extends that to longer texts. The Google Web 1T N-grams corpus contains the frequency count of English word n-grams (unigrams to 5-grams) computed over one trillion words from web page texts collected by Google in 2006.

The relatedness between two words is computed by considering the tri-grams that start and end with the given pair of words, normalizing their mean frequency with unigram the frequency of each of the words as well as the most frequent unigram in the corpus as shown in Figure 3.4, where $C(\omega)$ is the frequency of the word ω. $\mu_T(\omega_1, \omega_2)$ is the mean frequency of trigrams that either start with ω_1 and end with ω_2, or start with ω_2 and end with ω_1. $\sigma(a_1, \ldots, a_n)$ is the standard deviation of numbers $a_1, \ldots, a_n$, and $C_{\max}$ is the maximum frequency among all unigrams.

GTM computes a score between 0 and 1 to indicate the relatedness between two texts based on the relatedness of their word content. For given texts P and R where $|P| \leq |R|$, first all the matching words are removed, and then a matrix with the remaining words $P' = \{p_1, p_2, \cdots, p_m\}$ and $R' = \{r_1, r_2, \cdots, r_n\}$ is constructed where each entry is a GTM word relatedness $a_{ij} \leftarrow GTM(p_i, r_j)$.

$$M = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

From each row $M_i = \{a_{i1} \cdots a_{in}\}$ in the matrix, significant elements are selected if their similarity is higher than the mean and standard deviation of words in that row:

$$A_i = \{a_{ij} | a_{ij} > \mu(M_i) + \sigma(M_i)\},$$

where $\mu(M_i)$ and $\sigma(M_i)$ are the mean and standard deviation of row i. Then the document relatedness can be computed using:

$$Rel(P,R) = \frac{(\delta + \sum^m a_{i=1} \sigma(A_i)) \times (m+n)}{2mn}$$

where $\sum^m a_{i=1} \sigma(A_i)$ is the sum of the means of all the rows, and δ is the number of removed words when generating P' or R'.

3.5 Semantic Jaccard Coefficient

Jaccard similarity coefficient is a statistic used to compute the similarity and diversity between two sets. Chierichetti et al. [33] showed that finding an optimal solution for weighted Jaccard median is an NP-hard problem and presented a heuristic algorithm to

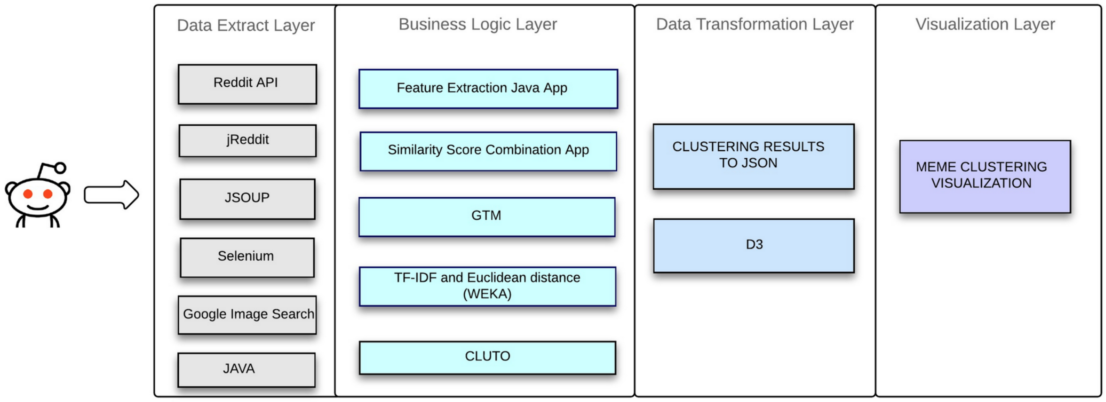

Data Extract Layer
Business Logic Layer
Data Transformation Layer
Visualization Layer
Reddit API
jReddit
JSOUP
Selenium
Google Image Search
JAVA
Feature Extraction Java App
Similarity Score Combination App
GTM
TF-IDF and Euclidean distance (WEKA)
CLUTO
CLUSTERING RESULTS TO JSON
D3
MEME CLUSTERING VISUALIZATION

Figure 3.5: The proposed meme detection framework.

speed up the computational complexity. The Jaccard coefficient between two sets A and B is defined as follows:

$J(A,B) = \frac{A \cap B}{A \cup B}$ where $0 \leq J(A,B) \leq 1$

We propose a modified version of Jaccard coefficient that exploits the use of semantic similarity using GTM. As the original Jaccard coefficient only uses an exact pattern matching, it does not work well if two Wikipedia concepts are not the same but are semantically similar. For example, the Jaccard coefficient for two concepts “President of the United States” and “Barack Obama” should be high as they are semantically similar using GTM.

For two submissions $S_1 = \{T_{11}, T_{12}, ..., T_{1n}\}$ and $S_2 = \{T_{21}, T_{22}, ..., T_{2n}\}$ where T_i is a Wikipedia concept extracted from the title or comments of submission S_i, the Semantic Jaccard Coefficient (SJC) is defined as:

$$SJC(S_1,S_2) = \frac{S_1 \cap S_2}{S_1 \cup S_2} \; where \; 0 \leq SJC(S_1,S_2) \leq 1 \tag{3.1}$$

where T_{1i} and T_{2j} are semantically equivalent, $T_{1i} \equiv T_{2j}$, if $GTM(T_{1i}, T_{2j}) \geq e$, where e is a parameter that is explored through the experiment. If T_i is semantically similar to more than one concept in S_2, we use the concept with the highest GTM score.

3.6 Similarity Scores and Combination Strategies

This section explores the use of GTM semantic similarity of Dang et al. [41] and introduces Wikipedia concepts as an external knowledge to propose five semantic similarity scores and their combinations between submissions. Representing a submission S in Reddit as a vector $S = (T,M,I,U,W)$ where:

- T is an n-dimensional feature vector $t_1, t_2, ...t_n$ representing the title of the submission and its description.
- M is an n-dimensional feature vector $m_1, m_2, ...m_n$ representing the comments of a submission.
- U is an optional n-dimensional feature vector $u_1, u_n, ...u_n$ representing the external URL content of a submission.
- I is an optional n-dimensional feature vector $i_1, i_2, ...i_n$ representing the image content of a submission. This content is extracted by using Google reverse image search, which takes an image as a query and extracts the text content of the website that is returned from the top search result and is not from Reddit.
- W is an optional n-dimensional feature vector $w_1, w_2, ...w_n$ representing the Wikipedia concepts of the titles and comments of a submission.

3.6.1 Similarity Scores

We propose five similarity measures between two submissions S_1 and S_2:

- **Title similarity** SC_t is the GTM semantic similarity score between the title word vectors T_1 and T_2.
- **Comment similarity** SC_m is the GTM semantic similarity score between the comment word vectors M_1 and M_2.
- **URL similarity** SC_u is the GTM semantic similarity score between the URL content word vectors U_1 and U_2.
- **Image similarity** SC_i is the GTM semantic similarity score between the word vectors I_1 and I_2 retrieved from Google Reverse Image Search.
- **Wikipedia similarity** SC_w is the SJC score between the bag of concept vectors W_1 and W_2 retrieved from titles and comments of submissions using Equation 3.1.

3.6.2 Combination Strategies

The main goal of this section is to study the effect of different similarity scores and their combinations on the quality of the meme clustering tasks. We incorporate Wikipedia concepts as an external knowledge to all combination strategies from our previous work [41].

3.6.2.1 Pairwise Maximization Strategy

The pairwise maximization strategy chooses the highest among the title, comment, URL, and image scores to decide the similarity between two submissions. This strategy avoids the situation where similarity scores have a low content quality (e.g., titles are short and lack details, comments are noisy, images and URLs are not always available) by choosing the most similar among them.

Given two submissions $S_1 = \{T_1, M_1, I_1, U_1, W_1\}$ and $S_2 = \{T_2, M_2, I_2, U_2, W_2\}$, the pairwise maximization strategy between them is defined as:

$$MAX_{S_1\,S_2} = MAX(GTM_{T_1T_2},\ GTM_{M_1M_2},\ GTM_{U_1U_2},\ GTM_{I_1I_2},\ SJC_{W_1W_2}) \quad (3.2)$$

where $GTM_{\mathrm{T_1T_2}}, GTM_{\mathrm{M_1M_2}}, GTM_{\mathrm{U_1U_2}}, GTM_{\mathrm{I_1I_2}}$ are the title, comment, URL, and image similarity scores between the two submissions S_1 and S_2. $\mathrm{SJC}_{\mathrm{W_1W_2}}$ is the SJC score between two submission S_1 and S_2 using Equation 3.1 for the Wikipedia concepts extracted from submission titles and comments.

3.6.2.2 Pairwise Average Strategy

The pairwise average strategy computes the average value of the five pairwise similarity scores. This strategy balances the scores among the five similarities in case some scores do not reflect the true content of the submission. It is defined as follows:

$$AVG_{S_1\,S_2} = AVG(GTM_{T_1T_2},\ GTM_{M_1M_2},\ GTM_{U_1U_2},\ GTM_{I_1I_2},\ SJC_{W_1W_2}) \quad (3.3)$$

3.6.2.3 Linear Combination Strategy

In the linear combination strategy, users can assign different weighting values manually. For example, if users think the title text does not capture the topic of a submission, they can assign a low weight factor (e.g., 0.1). If they think comment texts are longer and represent the topic better, they can assign a higher weight factor (e.g., 0.6). The linear combination strategy is defined as follows:

$$LINEAR_{S_1\,S_2} = LINEAR(w_tGTM_{T_1T_2},\ w_mGTM_{M_1M_2},\ w_uGTM_{U_1U_2},\ w_iGTM_{I_1I_2},\ w_wSJC_{W_1W_2}) \quad (3.4)$$

where w_t, w_m, w_u, w_i, and w_w are the weighting factors for titles, comments, images, URLs, and Wikipedia concepts with a normalization constraint $w_\mathrm{t} + w_\mathrm{m} + w_\mathrm{u} + w_\mathrm{i} + w_\mathrm{w} = 1$.

3.6.2.4 Internal Centrality-Based Weighting

Computing the optimized weight factors for the linear combination strategy is an intensive task. JafariAsbagh et al. [91] used a greedy optimization algorithm to compute the optimized linear combination for the task of clustering memes. However, it is unrealistic to compute all the possible weighting combinations for Equation 3.4. To alleviate this computational cost, we propose the Internal Centrality-Based Weighting (ICW), a novel strategy to automatically calculate the weight factors of the linear combination strategy. This strategy calculates the weight factors for each element of a submission by considering its surrounding context. Although all elements of a submission are semantically related, some elements could have more semantic content than others; for example, the URL content discusses more the topic than the title. More weight is assigned to the elements with higher semantic content. The proposed strategy is shown in Equation 3.5. It computes the semantic content weights using internal and external similarity scores between titles, comments, URLs, images, and Wikipedia concepts of two submissions. We append all the Wikipedia concepts together to compute the GTM score between Wikipedia concepts and other texts. For each submission, this strategy computes the centrality score for each element of each submission S_i :

$CENT_{\mathrm{T_i}} = GTM_{\mathrm{TM_i}} + GTM_{\mathrm{TU_i}} + GTM_{\mathrm{TI_i}} + GTM_{\mathrm{TW_i}}$
$CENT_{\mathrm{M_i}} = GTM_{\mathrm{MT_i}} + GTM_{\mathrm{MU_i}} + GTM_{\mathrm{MI_i}} + GTM_{\mathrm{MW_i}}$
$CENT_{\mathrm{U_i}} = GTM_{\mathrm{UT_i}} + GTM_{\mathrm{UM_i}} + GTM_{\mathrm{UI_i}} + GTM_{\mathrm{UW_i}}$
$CENT_{\mathrm{I_i}} = GTM_{\mathrm{IT_i}} + GTM_{\mathrm{IM_i}} + GTM_{\mathrm{IU_i}} + GTM_{\mathrm{IW_i}}$
$CENT_{\mathrm{I_w}} = GTM_{\mathrm{WT_i}} + GTM_{\mathrm{WM_i}} + GTM_{\mathrm{WU_i}} + GTM_{\mathrm{WI_i}}$

then, it computes the weighting factors between two submissions S_1 and S_2 by:

$w_{\mathrm{T}} = CENT_{\mathrm{T_1}} * CENT_{\mathrm{T_2}}$
$w_{\mathrm{M}} = CENT_{\mathrm{M_1}} * CENT_{\mathrm{M_2}}$
$w_{\mathrm{U}} = CENT_{\mathrm{U_1}} * CENT_{\mathrm{U_2}}$
$w_{\mathrm{I}} = CENT_{\mathrm{I_1}} * CENT_{\mathrm{I_2}}$
$w_{\mathrm{W}} = CENT_{\mathrm{W_1}} * CENT_{\mathrm{W_2}}$

then, it normalizes the weighting factors so that: $w_{\mathrm{T}} + w_{\mathrm{M}} + w_{\mathrm{U}} + w_{\mathrm{I}} + w_{\mathrm{W}} = 1$, and finally computes the ICW strategy:

$$ICW_{S_1 S_2} = ICW(w_T GTM_{T_1 T_2},\ w_M GTM_{M_1 M_2},\ w_U GTM_{U_1 U_2},\ w_I GTM_{I_1 I_2},\ w_W GTM_{W_1 W_2}) \quad (3.5)$$

3.6.2.5 Similarity Score Reweighting with Relevance User Feedback

One effective way to improve the clustering results is to manually specify the relationships between pairwise documents (e.g., must-link and cannot-link) to guide the document clustering process [16]. As social network data are intrinsically heterogeneous and multidimensional, it is not easy to compare two submissions to determine if they are similar or not without putting them into the same context. To overcome this limitation, a novel technique, the Similarity Score Reweighting with Relevance User Feedback (SSR), is proposed to incorporate relevance user feedback by a visualization prototype in which submissions are displayed as a force-directed layout graph where:

- **A node** is a submission in Reddit.
- **An edge** is a connection between two submissions if their similarity scores are above a threshold (default 0.85).
- **A node colour** represents to which cluster it belongs.

Algorithm 1 describes how the visualization system integrates user feedback to remove outliers, move submissions from a cluster to another, or reassign similarity score weighting factors for submissions. Users can select any of the five proposed strategies, MAX, AVG, LINEAR, and ICW as a baseline for clustering. Figure 3.6 (a) shows an SSR visualization

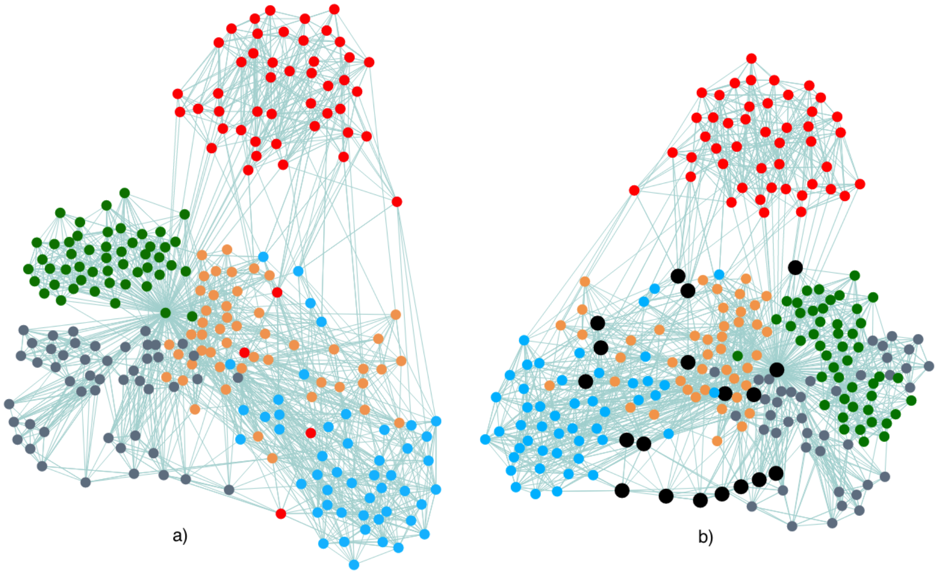


Figure 3.6: The proposed meme visualization: a) The original visualization graph and b) The clustering result using ICW

Algorithm 1 Semi-supervised Similarity Score Reweighting with Relevance User Feedback Strategy (SSR).

Input: a set of submissions X from Reddit.

Output: K clusters $\{X\}_{l=1}^{K}$

```
loop
    {Step 1} Perform k-means clustering on P percent of the ground-truth dataset using one of the proposed strategies. P is defined through experiments.
    {Step 2} Visualize the clustering result in step 1.
    {Step 3} Allow users to interactively remove outlier submissions, reassign submission class labels, or assign weight factors for each element between two submissions.
    {Step 4} Re-cluster the submissions based on user inputs.
    {Step 5} repeat step 1 if necessary.
end loop
{Step 6} Recluster the whole dataset considering user feedback in Step 1 to 5.
```

of the meme dataset using ICW strategy. The graph has five different colours that represent five memes in the ground-truth dataset. Users can pan, zoom, or click on a submission to get more details about this submission. They can also click on the checkbox “Show wrong cluster assignments” to see which submissions are incorrectly assigned by the ICW strategy. Based on the graph visualization, users can understand how a submission is positioned regarding its neighbour submissions. When clicking on a node in the graph, users will be redirected to the actual submission in Reddit to find out more information and decide if it belongs to the correct cluster. Most of the incorrectly clustered are overlapped or outlier nodes as shown in Figure 3.6 (b). For each incorrectly assigned submission, users can remove, update its class label, or assign a different similarity coefficient score for each element between two submissions. SSR focuses on human knowledge to detect outliers or borderline submissions.

3.7 The Offline-Online Meme Detection Framework

The meme detection problem is defined for any social media platform used to spread information. In these systems, users can post a discussion or discuss a current submission. An overview of the proposed meme detection framework is shown in Figure 3.5.

3.7.1 The Offline-online Meme Clustering Algorithm

As OSN data is changing and updating frequently, we modify and extend the proposed ICW algorithm [41] to work with the online streaming clustering algorithm using the semantic similarity and Wikipedia concepts to handle continuously evolving data over time. As emerging events or topics are changing in real time, some topics may appear but not burst. Other topics or events may appear and become a popular topic for a long period. Based on this observation, the proposed framework adopts the damped window model [66] and assigns more weight to recent data and popular topics. It also adopts the offline-online components of Aggarwal et al. [1] to make the online meme clustering more efficient. As clustering OSN data is a computationally intensive task, the offline component does a one-pass clustering for existing OSN data in the first step. It also calculates and summarizes each cluster statistics using Wikipedia concepts extracted from the titles and comments of all the submissions in the same cluster. For the online component, it assigns new data points to the appropriate clusters using a modified version of online k-means.

For an online stream $S_1, S_2, ..., S_n$ where each S_i is a submission in Reddit. Each submission S_i is represented by a 5-tuple (T, C, U, I, W) that represents the title, comments, URL, image, and Wikipedia concepts (of titles and comments) of this submission. At a time t, the proposed framework are presented in two steps:

Table 3.1: The experiment ground-truth dataset.

No.	Topic	Submission Counts	Comments	Submissions with a URL	Submissions with an Image
1	EBOLA	495	89394	218	39
2	FERGUSON	495	83912	203	87
3	ISIS	488	76375	190	61
4	OBAMA	490	139478	142	13
5	Trayvon Martin	471	93848	250	30

- **Offline component:** cluster all the submission from $t-1$ to t_0 into k clusters $C_1, C_2, ..., C_n$, such that.
  - Each submission S_i belongs to only one cluster.
  - The submissions are clustered into clusters using the ICW strategy.
- **Online component:** assign an incoming submission S_i into one of the clusters created from the offline component.

The online k-means and the sliding window model of JafariAsbagh et al. [91] do not consider the popularity and occurrence frequency of a topic. To overcome this problem, we proposed an approach to compute the popularity of topics and use it as a parameter for the damped window model. Each cluster is represented by a set of Wikipedia concepts w_1,w_2,...w_n and each concept can be linked back to the original submissions. Each concept in the cluster statistics is represented by the exponential decay function:

$$W(t) = N * e^{-\lambda t} \tag{3.6}$$

where N is the count of this concept in the cluster. λ is a positive exponential decay constant. If a concept stays in the cluster for a period but there are no new submissions that contain this concept, it will be removed from the cluster set. A summary of the proposed algorithm is shown in Algorithm 2 and its explanation is shown in Figure 3.7.

For the online component, when a new submission is assigned to a cluster, it may not naturally belong to this cluster. For example, this submission can be an outlier or the beginning of a new cluster. Equation 3.1 handles these two situations naturally. If the submission is an outlier, it will be removed afterward if there are no similar incoming

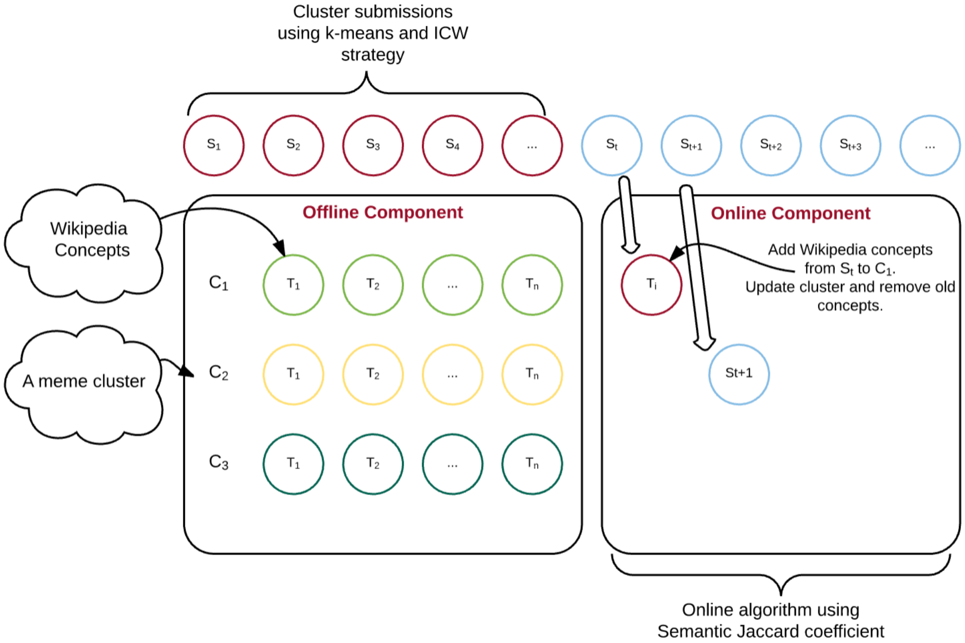


Figure 3.7: The offline-online meme clustering algorithm.

Algorithm 2 The Offline-online Meme Clustering Algorithm

```
Input: a set of submissions $S = \{S_1, S_2, ..., S_n\}$ from Reddit at time t. k = number of clusters.
    m = number of concepts in each cluster.
Collect all submissions in Reddit for an intial period of time t.
Cluster all the collected submissions using ICW strategy into k clusters $C_0, C_1, ..., C_k$.
Summarize each cluster by extracting Wikipedia concepts from titles and comments.
loop
    Retrieve the next submission $S_{t+1}$.
    For each summarized cluster $C_i$, compute $SJC(S_i, C_i)$ score.
    Assign submission $S_{t+1}$ to the cluster with the highest SJC score.
    Re-compute the cluster summary statistics of the selected cluster.
    Only keeps m concepts in each cluster using Equation 3.6. If a submission has no
    existing concepts in the summary, remove this submission from the cluster.
end loop
```

submissions that contain concepts from this submission. For the second situation, the cluster will be naturally replaced by the new concepts and the old concepts will be removed.

3.8 Experimental Results

The objective of this section is to evaluate the performance of the meme clustering tasks with the incorporation of Wikipedia concepts and the proposed online meme clustering algorithms. We re-evaluate the first three experiments from our previous work [41] with the incorporation of Wikipedia concepts and further access three new experiments. First, we explain how the ground-truth dataset is extracted from Reddit, and then discuss the evaluation metric and how the experiments are carried out.

3.8.1 Ground-truth Dataset

To study the spread of memes in Reddit, the posts and comments related to a specific meme are identified. A generic query is used to capture all of the related submissions for a specific meme. Since there are no available Reddit meme datasets, RedditAPI and jReddit, an open source Java project, are used to extract submissions, comments, and other data views (image and URL content) about a specific meme using predefined regular expressions. All submissions that do not have any comments or “up” or “down” votes are removed, as we assume that users are not interested in them. In addition, comments less than 5 words long are ignored. Stop words are also removed. We also manually remove the submissions that seem not to be related to the search topic. For each submission with a URL in its title, JSOUP is used to parse the main body text content of the URL. Occasionally, a submission can have an image in its title. Selenium is used to submit the image to Google Reverse Image Search to find the most similar webpage to this image. If the top-searched result returns an article from Reddit, the program traverses through the search result list until it finds an article that is not from Reddit. Wikipedia concepts are extracted using Dexter [28].

The ultimate goal of this framework is to detect memes and discussion topics online. In order to access the performance of the proposed similarity strategies, we collect ground-truth data for the experiments. First, the five most popular topics in Reddit from October to November 2014 are selected. The program extracts titles, comments, URL, and image content of all related submissions for each topic. Each topic is labeled to the corresponding cluster based on the keyword search. The five topics (clusters) are: (1) EBOLA (2) Ferguson (3) ISIS (4) Obama and (5) Trayvon Martin. Table 3.1 shows the detailed statistics of the ground-truth dataset.

3.8.2 Clustering Algorithms

The paper adopts k-means clustering as the base clustering algorithm because of its simplicity and efficiency. The experiments focused more on determining if the proposed similarity strategies improve clustering results and the proposed online clustering algorithm could detect emerging memes in real time. GTM is used to compute the similarity score between texts. The output of the GTM algorithm is a similarity matrix that shows the similarity score for each text with the other texts in the dataset. After producing this similarity matrix, gCLUTO is used to cluster the matrix using an equivalent version of k-means clustering.

3.8.3 Baselines

For the baselines, each title, comment, URL, or image text is represented as a TF-IDF vector. Euclidean distance [54] is used to calculate the similarity score between TF-IDF vectors due to its simplicity. In the next section, we compare the clustering results of the proposed strategies and algorithms with the baseline.

3.8.4 Results

For the ground-truth dataset, since the class labels exist for all of the submissions, purity is adopted (i.e., the number of correctly assigned submissions over the total number of submissions) as an evaluation measure. Larger purity value indicates better clustering results. Several configurations are explored to evaluate the performance of the proposed similarity strategies. As URL, image content, and Wikipedia concepts are not always available, they are used as additional data for the clustering tasks for MAX, AVG, and ICW. The proposed configurations are used for both GTM similarity and baseline similarity and configured as:

- **TITLE**: Only use the title similarity for pairwise submission comparison.
- **COMMENT**: Only use the comment similarity for pairwise submission comparison.
- **MAX**: Use the maximum of the five similarity scores for pairwise submission comparison as defined in Equation 3.2.
- **AVG**: Use the average of the five similarity scores for pairwise submission comparison as defined in Equation 3.3.
- **ICW**: Calculate the pairwise similarity between two submissions based on internal centrality weighting as defined in Equation 3.5.

3.8.4.1 Clustering with Semantic Similarity Scores

The first experiment explores the advantage of using GTM semantic similarity and Wikipedia concepts for meme clustering tasks. The k-means clustering results between the proposed similarity scores and the baselines using TF-IDF and Euclidean are compared for TITLE, COMMENT, MAX, and AVG. GTM score outperforms the baselines as shown in Figure 3.8 and 3.9. Using comment content for clustering produces a better result than using title content as title texts are usually concise and do not represent the context of a submission. Exploiting additional image and URL content by AVG and MAX strategies improves the clustering results as shown in Figure 3.10 and 3.11. Another interesting result is that the performance of GTM for comments is very close to the AVG strategy. Using AVG strategy does not capture the semantic content of each similarity score efficiently. The experiment results also show that GTM & Wikipedia score scales better than the baseline for higher vector dimensions. We conjecture that GTM & Wikipedia helps alleviate "the curse of dimensionality" for clustering using traditional similarity measures.

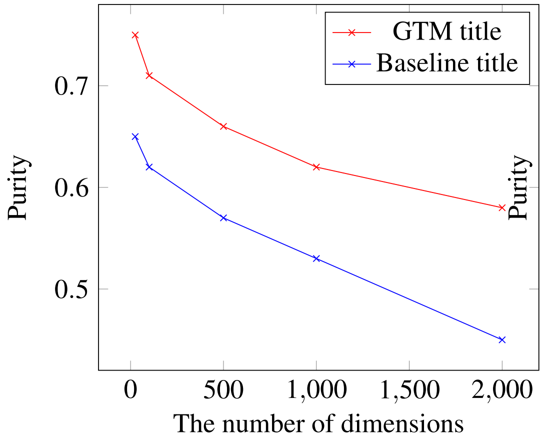


Figure 3.8: GTM title vs. Baseline title

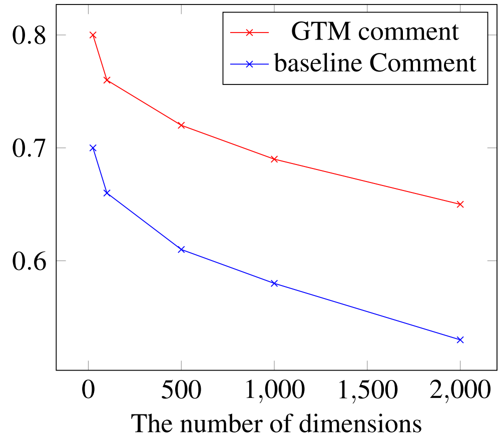


Figure 3.9: GTM comment vs. Baseline comment

3.8.4.2 The ICW Strategy

In this experiment, the objective is to find out if the proposed ICW strategy with the use of GTM and Wikipedia concepts improves the clustering result for a meme clustering task. The experiment results between the proposed ICW, AVG, and MAX are shown in Figure 3.12. Results indicated that ICW outperforms AVG and achieves better results than MAX. We also found that the AVG combination does not provide good results when comparing with using MAX or ICW. One of the reasons may be each similarity score plays

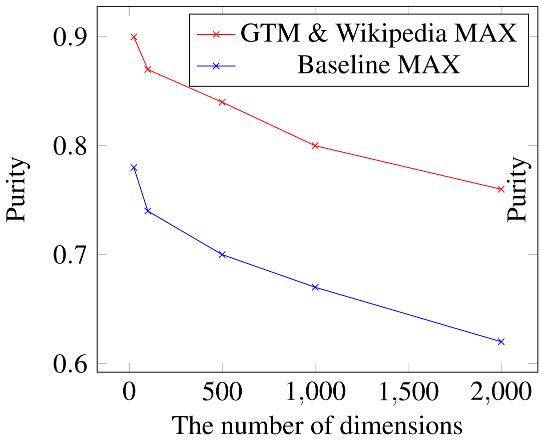


Figure 3.10: GTM & Wikiepdia MAX vs. Baseline MAX

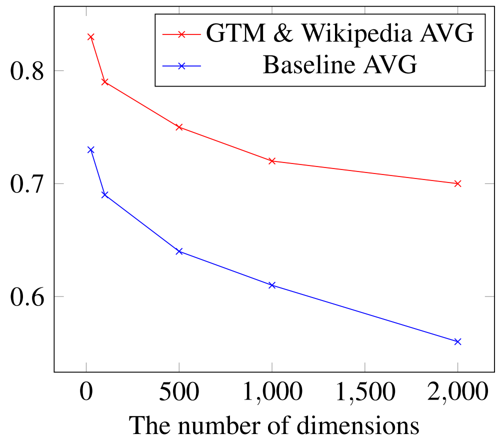


Figure 3.11: GTM & Wikipedia AVG vs. Baseline AVG

a different role in distinguishing memes in Reddit and this agrees with our assumption about the semantic content related between elements of submissions.

3.8.4.3 Similarity Score Reweighting with Relevance User Feedback

This section investigates the improvement from using user feedback with the visualization prototype for the meme clustering task. At first, users can select one of the five proposed strategies (AVG, MAX, LINEAR and ICW) to cluster the ground-truth dataset. For the LINEAR, we explore different weight factors for the Equation 3.4. Although the clustering results are improved when weight factors for title and comment are low (e.g., 0.1 for titles, 0.3 for comments) and are high for URLs and image (e.g., 0.6), their results are still not optimized when comparing with MAX and ICW. We remove outliers and reassign the weight factors for overlapping nodes. The clustering results are statistically improved for both MAX, AVG, and ICW at p = 0.05 as shown in Figure 3.12 (SSR ICW, SSR MAX, SSR AVG).

3.8.4.4 Similarity Score with Wikipedia Concepts

This section studies whether the use of Wikipedia concepts improves the clustering results. We compared the clustering results with the three combination strategies (MAX, AVG, and ICW). We also explore whether the use of titles and Wikipedia concepts alone produces a good clustering result. The clustering results are statistically improved for MAX, AVG, and ICW at p = 0.05 as shown in Figure 3.13. We also observed that using only titles and Wikipedia concepts did not achieve good clustering results. One of the underlying reasons is because there are overlapping concepts between memes which downgrades the

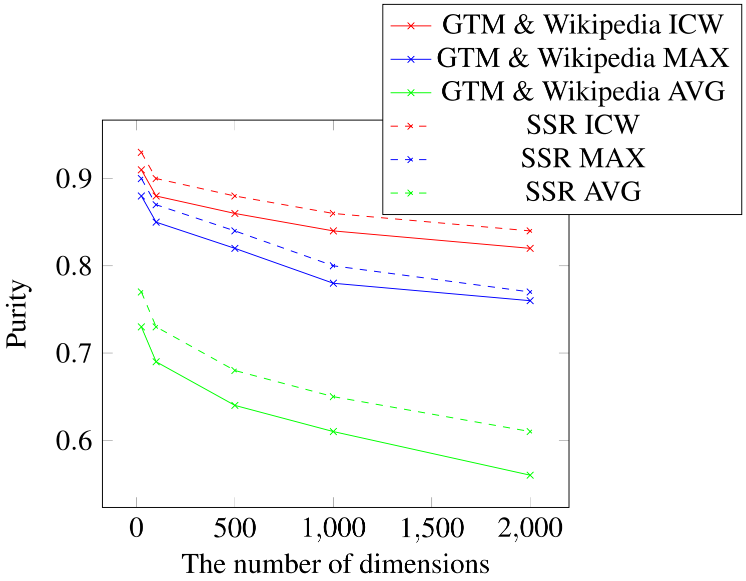


Figure 3.12: Clustering results with different similarity score strategies.

clustering results.

3.8.4.5 Semantic Jaccard Coefficient

This section inspects whether the proposed Jaccard coefficient helps to increase the accuracy of the meme clustering results. We investigate whether the semantic Jaccard coefficient increases the clustering results of MAX and AVG ($e = 0.8$). We find that Jaccard coefficient using semantic similarity significantly increases the clustering results, as show in Figure 3.14. This happens because there are many Wikipedia concepts that are very similar but not the same in the ground-truth dataset, for example, “Obama” vs. “President”. Combining the Wikipedia concepts with the semantic similarity scores using SJC significantly improves the clustering results.

3.8.4.6 The Offline-Online Meme Clustering Algorithm

This section examines whether the proposed offline-online clustering algorithm achieves a good clustering result. We use the ground-truth dataset to evaluate the algorithm. For the offline component, we select the first 2000 submissions from the ground-truth and cluster them into 5 clusters using the ICW strategy. For each cluster, we extract all the Wikipedia concepts from titles and comments of each submission as the cluster summary. We process the remaining 500 submissions using the online component in an ascending time order. Each incoming submission is assigned to its closest cluster using Equation 3.6 ($\lambda = 1$). Finally, the clustering accuracy results between the offline-online clustering algorithms and ICW

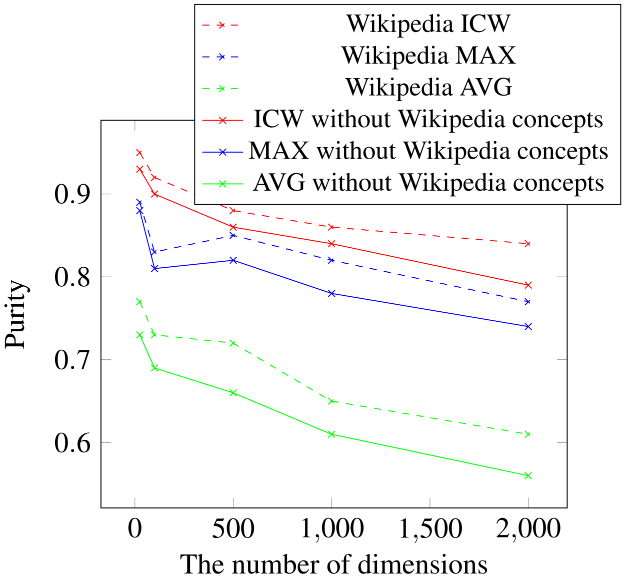


Figure 3.13: Clustering results with Wikipedia concepts.

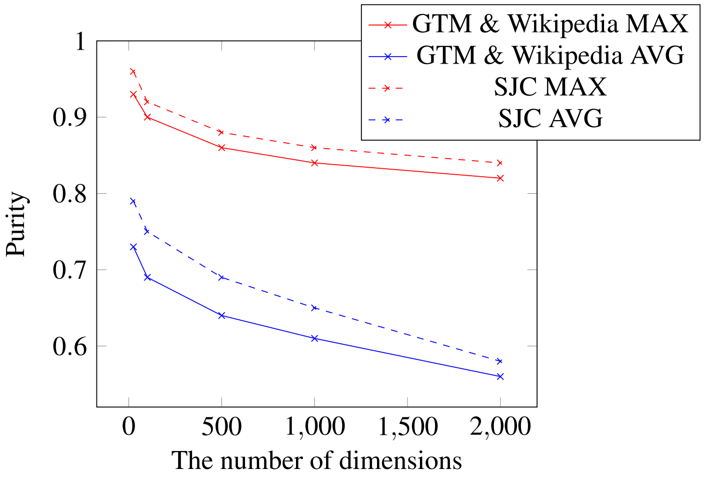


Figure 3.14: The proposed modified Jaccard coefficient vs. MAX and AVG

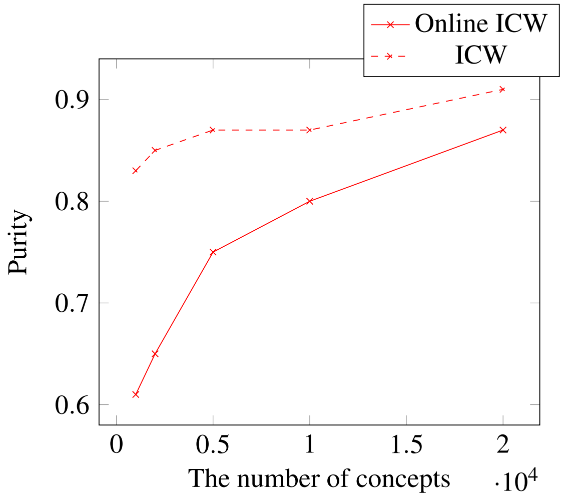


Figure 3.15: The proposed online clustering algorithms vs. ICW

are shown in Figure 3.15 where the x-axis represents the increasing number of concepts. The experiment results show that although the proposed online clustering algorithm did not perform as well as the ICW with a low number of concepts, the results are comparable when the number of concepts is higher than 10000. In addition, although using only Wikipedia concepts does not work well for the offline mode, it achieves a reasonable accuracy for the offline-online algorithm. The underlying reason is that we use the offline component as a learning model using ICW strategy, and it helps alleviate the noisy nature of OSN data.

3.9 Conclusions

This chapter presents an online framework to tackle the problem of meme clustering in Reddit as a means to detect emerging events and rumour-driven topics and their spread as specific clusters. This framework makes use of Google Web 1T n-gram corpus (GTM algorithm) to compute the similarity between texts and Wikipedia concepts as external knowledge for the meme clustering task. It also defines several pairwise similarity scores between elements of two submissions. These scores include external content related to image and URL elements of a submission. The paper explores a semantic similarity version of Jaccard coefficient and several strategies to combine the similarity scores in order to produce better clustering results. These strategies include average, maximum, linear combination, internal centrality-based weighting, and similarity score reweighting with relevance user feedback. Finally, it proposes an offline-online meme clustering framework to both detect memes in real time and achieve good clustering results.

The experimental results demonstrate that using GTM semantic similarity improves the clustering results compared to the baselines. Using Wikipedia concepts as external knowledge also helps increase the accuracy of clustering results. In addition, the Similarity Score Reweighting with Relevance User Feedback strategy achieves the best result and the Internal Centrality-Based Weighting strategy performs better than AVG and MAX, as the first strategy allows users to assign different similarity scores for different elements between two Reddit submissions and the second strategy computes the weighting factor of each element of a Reddit submission based on its semantic content. The offline-online clustering algorithm achieves a comparable result to the ICW when the number of concepts is large in the cluster summary.

In future work, we aim to extend the proposed framework to other social network websites, such as Twitter, Facebook, and Google Plus. Another important direction is to extend this framework for studying the spread of rumours in online social networks, for example, visualizing how a rumour-related meme is discussed and spread in Reddit. This will help researchers to understand the patterns of how a rumour is spread, its pattern and detect emerging rumours. Comparing the spread of rumour-driven memes between Reddit and other OSNs and finding a correlation between them will provide a more holistic view of rumour spread.

Chapter 4

RumourFlow: A Visual Analysis Framework for Studying Rumor Spread Patterns and Influencing Users in Social Media

With the rapid increase of rumor propagation in Online Social Networks (OSNs), analyzing, detecting and understanding characteristic patterns, evolution and user behaviors behind this activity is essential. However, there is little work that studies and supports the in-depth analysis of rumor spread and activities of players in their propagation. In this chapter, we propose RumorFlow, a visual analysis framework to explore rumor spread patterns and life cycles as well as their association with influencing users. RumorFlow provides visualizations built on state-of-the-art techniques and implements data analysis models to reveal rumor contents and participants' activity, both within a rumor and across different rumors. The system also employs various information diffusion models and text mining techniques, such as sentiment analysis, Wikipedia entity linking, and text semantic similarity for visualizing rumor strength, topic evolution, user participation in studying this phenomenon. The effectiveness of the approach is illustrated by use cases highlighting relevant insight into rumor spreading, most of which could not otherwise be observed, in conformity with what was revealed by the analysts' opinions. Qualitative studies from experts confirm the effectiveness of RumorFlow in drawing hypotheses and patterns regarding rumor propagation. The basis of this chapter is from the published paper [44].

4.1 Introduction

A rumor is generally defined as a truth-unverifiable statement that appears and is disseminated in uncertain situations [137]. Rumors arise quickly when their themes are interested to a large number of individuals, and their truths are not straightforward to verify [56]. Online Social Networks (OSNs) have recently emerged as the favored media for spreading information such as breaking news, sport events, fashion trends [95], as well as rumors and other less credible information, with unverified authority and sources [81]. As an example, so many rumors about the Swine Flu outbreak in 2009 were communicated via OSNs that the US government had an official website to debunk and limit the spread of these rumors [119, 31].

Many researchers have actively investigated how to detect rumors efficiently in OSNs [132, 136, 107, 151], mostly trying to detect rumors in OSNs by applying different feature sets using a supervised learning approach. However, this method does not work well due to the morphing characteristics of rumors and rumor characteristics can be different in various sources (e.g., Twitter vs. Reddit) or categories (e.g., politic vs. non-politic rumors). This makes detecting rumors a challenge unless the phenomenon is understood more deeply. For example, it is not trivial to determine when a rumor starts and how it ends. Each rumor can peak, get debunked, stay dormant, or disappear [92]. Additionally, user behavior towards rumors vary. As social network data is always changing and updating in real time, presenting time varying data properly is necessary (see, for instance, [7, 65]). There are various stochastic spread models [38, 111] to simulate how rumors are propagated between users in social networks. Also, researchers have tried to analyze and understand who spreads rumors [121, 122]. However, associating content flow, interest evolution, and user participation is necessary to completely describe (and thus better detect and influence) rumor spreading.

This chapter is an extended version of our previous work published in Dang et al. [44] which integrates the use of stochastic rumor spread models and visualization techniques to display how rumors are spread with regards to its popularity, topics, and user interactions. We introduce new visualization views that support analysts to analyze, detect, and microscopically examine the detailed characteristics of a rumor and how users interact with each other inside a rumor. The extended framework introduces an approach to tackle this multi-component problem by adapting state-of-the-art visualization techniques and providing a Visual Analytics (VA) tool for understanding and analyzing how rumors are disseminated and discussed by OSN users. It includes a set of analytical tools including rumor spread models, sentiment analysis, Wikipedia entity linking, and text semantic similarity, to help attain a visual summarization of the many facets of rumor spread, and provide various levels of granularity through an exploratory interface. The focus is on giving support to analyze past and present rumor flows as well as their players, so as to enable improvement of current models for rumor initiation and propagation, and for future rumor detection and fact checking algorithms.

The proposed RumorFlow system design principles are guided by two main observations from experts in rumor spread [56, 87]:

- Topics about a rumor evolve over time. Some new topics may appear, and others may fade.
- Most rumors are spread by comparatively few active users.

Focusing on these principles, the adopted information diffusion models, visual encoding and interactive techniques of the system aim to provide answers to the following research objectives:

- How can analysts observe rumor evolution over time?
- What players are key to sustaining and propagating rumors, and when do they act?
- Can analysts use the system to have a better understanding of rumor spread patterns and characteristics?

As a result, this work presents the following contributions:

- Extend the work of our research article [44] by designing and introducing more visualization views that coordinate multiple representations of rumor propagation and user activities for a specific rumor.
- Integrate information diffusion models, social science theories, social network analysis, sentiment analysis, and text mining techniques to facilitate data exploration and analysis for online rumor spread. This completes previous work by providing a holistic analysis framework for rumor analysis.
- Design and evaluate case studies about eight widespread rumors in Reddit that qualitatively validate the effectiveness of the proposed visualizations and models.

To the best of our knowledge, RumorFlow is the first visual analytics system for exploring rumor spread with regards to topic flow and user interaction in social media derived from social science theories.

4.2 Related Work

This section presents current research on visualizing information flow and rumor information in OSNs, from which we draw connections to the work presented here.

4.2.1 Visualizing Information Flow

One of the first analysis tasks in our case is to examine topics communicated by participants' messages. Topic modeling and visualization have been intensively studied for detecting changing patterns in document collections over time.

ThemeRiver is a well-known approach to reflect patterns in temporal thematic changes in large collections of documents over time [79]. Each topic (theme) in the collection is mapped to a “current” in the “river”, and the “current” flows along with time from left to right. The width of a “current” at a specific time reflects the importance of themes, while the “river” width represents the global strength of all the themes. TextPool summarizes and visualizes emerging terms from the web as a graph displayed by force-directed placement, plotting similar terms close together [5]. The edge length between terms denotes their content similarity. Nodes and edges evolve over time as new terms appear and old terms become out-dated. Their system also animates the arrival of a new term so that end-users can notice it easier. STREAMIT presents a real-time interactive visualization to continuously illustrate growing document collections [7]. When a new document joins the corpus, the visualization will display a node-diagram graph using forced-directed placement. It helps end-users detect document outliers or emerging key terms by extracting keywords from documents using LDA (Latent Dirichlet Allocation), a commonly used topic extraction algorithm.

LDA is also used in TopicNets [72], an interactive web-based visualization application, to help learn the topical representation of each document in a corpus visualized as a node-link graph. This system provides two main views: (1) Topic-similarity layout and (2) Order-preserving layout. The first layout visualizes documents based on topic similarity, while the second orders documents according to specific attributes such as creation time or category. Termite visualizes relationships between documents in a corpus as a tabular matrix representation where columns represent documents, and rows show the keywords (topics) extracted by LDA [36]. A circle is used to represent the similarity between topics and documents in each matrix item.

All these approaches are very useful to understand topic presence, strength, and progression in a document set. However, for our application, LDA is not ideal in terms of topic detection, since communication in the form of microblog discourse is very noisy and the algorithm is also time-consuming. Additionally, individual documents in rumor spreading are less relevant than collective aggregations, such as aspects of contents over time, with individual messages being revealed on demand. The amount of individual messages for a rumor renders practically any point by point visualization impractical. This motivates us to adopt a Wikipedia entity linking approach which is a fast and simple way to extract topics from social media data. Current approaches also fall short of jointly examining user behavior and topic change in unison.

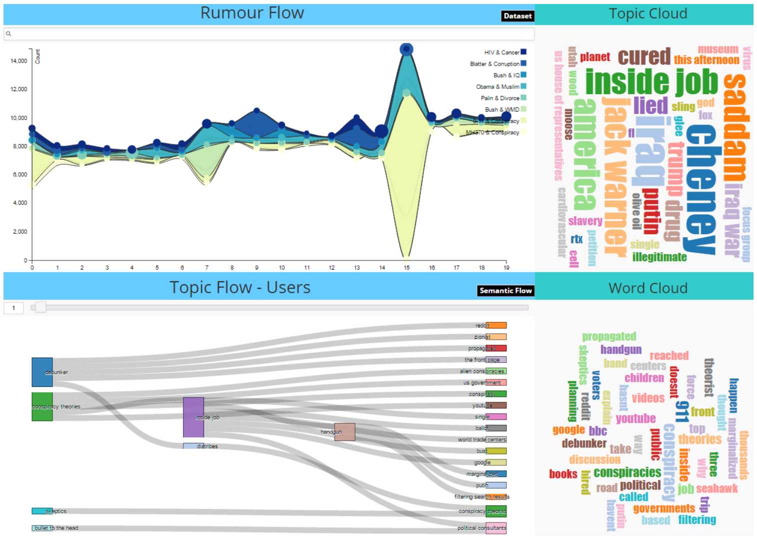


Figure 4.1: RumorFlow visualization interface for 8 rumors and $k = 20$. The top part is the current overall view, including the evolution of 8 rumors and concept word clouds. The bottom part details some aspect of the data, in this case, the topics extracted for one particular rumor, as well as rumor word clouds.

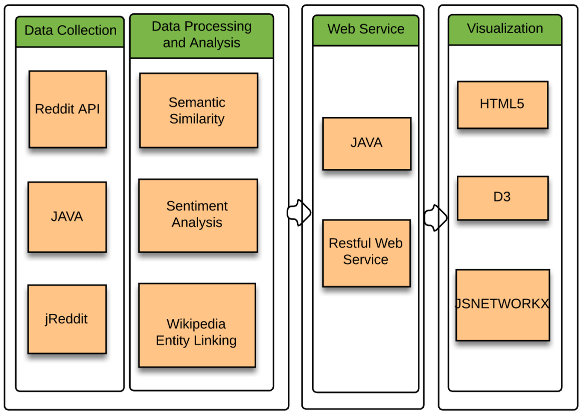


Figure 4.2: RumorFlow proposed visual analysis framework.

4.2.2 Visualizing Information and Rumor Flow in OSNs

Important previous work has also dealt with presenting information from social media. Word clouds of Twitter topics is visualized in a horizontal time series axis [133]. It uses LDA to extract the most recent 200 topics from Twitter in real time. This system allows users to make sense of Twitter content, together with other attributes, such as username, status, and

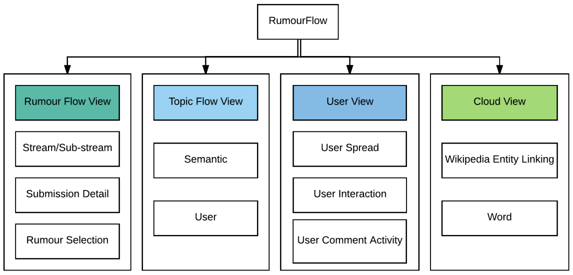


Figure 4.3: RumorFlow Visualization Views. Each main view includes many sub-views and each sub-view includes user interaction and search features.

user location. TwitterScope monitors and visualizes streaming messages from Twitter in real time [65]. The authors also introduced the use of a map metaphor and proposed a novel graph-packing algorithm to avoid node overlaps and underlaps. Tweets are represented as a map, where each tweet is represented as a “city” and similar tweets are grouped into a “country”. When a new tweet arrives, the system will recalculate the layout so that the current layout undergoes only minimal changes, in order to preserve the user mental map. Convis visualizes user comments in online conversations [86]. This system applies topic modeling techniques and sentiment analysis to help end users navigate and identify the most important topics and opinions expressed in the comment conversations.

OpinionFlow utilizes information diffusion theories to visualize how opinions are disseminated by Twitter users through the adoption of Sankey diagram and a tailored density map [157]. Users can select multiple topics from a stacked tree of topics to compare their diffusion patterns on Twitter. This last work is related to ours in that the flow of information (in this case, opinion) is central to the task. However, focus is first turned to calculating topics, and then examining chosen topics over time, where changes in topics are modeled by users from one topic influencing the other. In our case, the whole rumor has to be the central aspect, and the topics secondary to the main examination of rumor life cycle and participation. Also, influence is a matter of exploration for rumors, and not currently modeled by any validated model.

Some recent scholarly and industry-led projects have relied on visualizations to show information regarding rumors in OSNs. Truthy is a supervised-learning visualization framework to identify misleading political campaigns by collecting, analyzing, and visualizing messages from Twitter [135]. Similarly, rumors about the 2011 UK riots spread on Twitter are identified and visualized [39]. The system group related Twitter messages into the same cluster. FluxFlow is an interactive visualization system to analyze and discover anomalous information in OSNs [165]. The system visualizes Tweets and adopts a supervised learning approach to detect Tweet anomaly. Current work on rumors focuses on providing tools to support the detection stage of the process.

RumorFlow differs from previous approaches both in focusing on rumor lifecycle as well as in integrating information diffusion and user interaction views to provide a complete view of rumor spread in OSNs. It assumes rumor is detected since there are currently tools that can be used to retrieve them. Rumorflow provides a robust visual analytic tool so that individuals can explore in order to understand and associate different aspects of rumors.

4.3 RumorFlow Framework

The main layout of RumorFlow VA system is shown in Figure 4.1. It includes four main components: *Rumor Flow* view, *Topic Flow* view, *User Flow* view, and *Topic/Word Cloud* view. RumorFlow's internal components are illustrated in Figure 4.2 and a summary of the system visualization views is shown in Figure 4.3. The description of the each component, as well as the main visualizations associated with them, are presented next. We first describe the data collection strategy. Then, as the proposed framework focuses on visualizing how rumors evolve and spread in term of topics and user activities, we outline how we adopted various models of rumor spread (Sec. 4.3.2), topic flows (Sec. 4.3.3) and the way users interact with each other (Sec. 4.3.4).

4.3.1 Collecting Rumors

RumorFlow explores two sources of information: (1) a corpus of known rumors, and (2) social network data of rumor spread. It uses `Snopes.com` as a source for exploring and validating rumors, and `Reddit.com` as a source of conversations regarding rumors. Snopes is a website that collects memes, urban legends, and stories with unknown or uncertain origins. Reddit is a social networking and news website where users, called Redditors, can create a submission with text content and direct links to other online content. Other Redditors can comment on and vote to decide the rank of submissions. Reddit has many subcategories, called subreddits, which are organized by areas of interest. The site has a large user base that discusses a wide range of topics such as politics and world news every day. Every Reddit submission has the following: a title, which summarizes the topic of the submission, the full chain of comments posted on that submission and their users, and a possible link to an external source of related information (such as news articles).

Unlike other OSNs, Reddit implements an open data policy; hence, users can query any posted data on the website. Reddit starts to be a popular source for analyzing and studying social network data [42, 14]. This makes Reddit a valuable resource to study the spread of rumors in OSNs.

4.3.2 The Rumor Spread Models and Visualizations

For the spread of rumors, both the evolution (or flow) of information, the topics discussed by the players, and the user activities are of importance. This section describes how the three research objectives are handled and visualized in RumorFlow.

4.3.2.1 Rumor Flow Models and Visualization Views

One of the first roles of a visual system for rumor analysis is to visualize the rumor strength, as well as its main topics, and sentiment directions (see Objective 1). A rumor R is composed by a series of submissions $\{S_1, S_2, ..., S_n\}$, where S_i represents the submission at time t_i in Reddit. Each submission S is represented as $\{T, U, C\}$ where T is the set of topics $\{T_1, T_2, ...T_m\}$ which is extracted using Wikipedia entity linking from titles and comments of submissions, U is the set of users $\{U_1, U_2, ..., U_o\}$, and C is the set of comments $\{C_1, C_2, ..., C_p\}$ on the submission related to this rumor.

For the rumor spread visualization, RumorFlow adopts the rumor spread theory [56] where the significance of rumor spread is represented by the conceptual relationship:

$Rumor = Importance + Ambiguity$

This relationship is to define that a rumor popularity depends on its importance and ambiguity. The proposed system uses sentiment analysis to reflect the ambiguity or controversy of rumors. Researchers have shown that, for a blog post, the more controversial the comments, the more popular this post will be [49]. As the submission title is very short and concise, Wikipedia entity linking [149] was used to extract important topics at a time t. We model the rumor spread at time t as a tuple:

$Rumor(t) = Sentiment(S_t) + Topics(S_t) + Users(S_t)$

where, $Sentiment(S_t) = (Pos - Neg)/(Pos + Neg)$, and for a submission S_t, Pos = total number of comments with positive sentiment, Neg = total number of comments with negative sentiment. $Topics(S_t)$ is a set of topics extracted from the title using Wikipedia entity linking, and $Users(S_t)$ is a set of users that have comments in the submission at time t. The relationship above means that the system visualizes the spread of a rumor based on three parameters: the sentiment of its users, topics, and a number of users.

The *Rumor Flow* view (see top left part of Figure 4.1) presents an overview of how a rumor evolves over time. It places the different rumors in layers, or streams, adopting the "river" metaphor [79]. That view is created selecting the *k* most relevant posts using Reddit search by relevance in the life cycle of a rumor and plotting them in a temporal order. Thus the horizontal axis represents temporal order. The layer width codes the number of comments for that main post, as a measure of popularity. Each post is marked by a circle, whose size represents ambiguity (larger for more controversial).

In each stream, there is an internal layer, or sub-stream (see Figure 4.4), that reflects the number of unique users of a submission at a given time. A rumor with a high number of comments and high number of unique users will be more popular than a rumor with a high number of comments but a low number of unique users. Each main stream may also show, for each submission, either the topics extracted using Wikipedia entity linking or the user with the most comments. This view allows analysts to compare different rumors with regard to their life cycles, active players and popular topics from start to end. If only one rumor is selected, the x-axis shows the actual timeline of its submissions. Figure 4.4 shows the fluctuating nature of the rumor "MH370 Conspiracy" from 2014 to 2015.

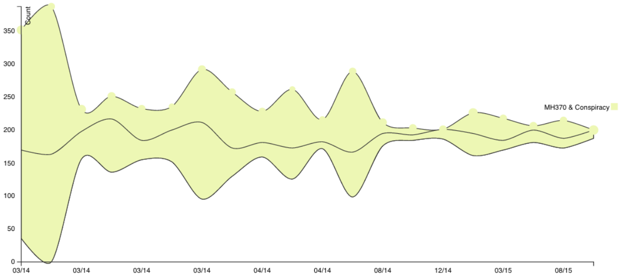
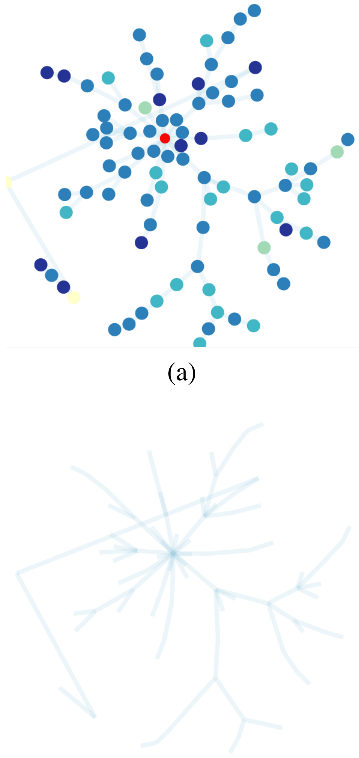


Figure 4.4: The fluctuating nature of the rumor "MH370 Conspiracy" from 2014 to 2015. The black line inside a stream (a sub-stream) represents the relation between the number of unique users and comments of a submission.

4.3.2.2 Additional Supporting Submission Views

The prototype that implements the RumorFlow approach with its models adds a series of additional representations that depict central aspects of rumors and user actions.

The *Comment Tree* view is a radial tree visualization for the comments to a particular post of interest. The available posts are the ones represented in the *Rumor Flow* view. In the tree, a red root node represents the original submission and every other node is a comment, in the comment chain. Initially, each level of the tree has a different color. Although the view is a simple tree, from the alternative visual codings various observations can be drawn, for instance:



(a)

(b)

Figure 4.5: An example of semantic and tree visualization view of a submission. Each node represents a comment, and the root (red) node is the original submission. (a) The strength of a node presents its semantic similarity between that node with its parent node. (b) The tree structure without nodes.

- *Color Coding* by similarity view (see Figure 4.5 (a)) allows identifying by changes in color comments that depart from the original posts. Thread hijackers or spammers (e.g., users that do not focus on the main topic) can be detected that way.
- *Tree* (see Figure 4.5 (b)) view shows the tree structure (without nodes) of a submission. The comment tree structures of two submissions can be visualized and compared.
- *Time-related* animation view adds comments in order of appearance on the tree. It is very useful to detect what comments generated interest at different times.
- *User and Comment* views plot usernames or comments on top of the nodes so that, under zoom, one can examine the reply chain in detail.

Table 4.1: The collected rumor datasets (number of submissions k = 20).

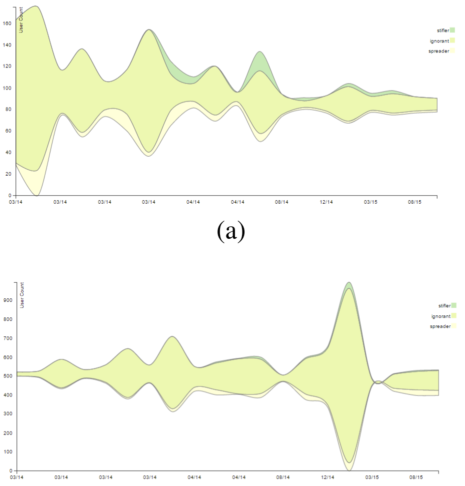
No.	Rumor	Keywords	No. Comments	No. Users	Start Date	Debunked Date
1	HIV can cure cancer	HIV & Cancer	3358	1112	2009	Unverifiable
2	Sepp Blatter Corruption Scandal	Blatter & Corruption	6062	1886	2014	Unverifiable
3	Bush has the lowest IQ	Bush & IQ	2162	1593	2001	2007
4	Obama is a Muslim	Obama & Muslim	12955	4112	2007	2009
5	Sarah Palin Divorce	Palin & Divorce	1272	348	2008	2014
6	Bush lied about WMD	Bush & WMD	4189	1068	2003	Unverifiable
7	9-11 Conspiracy	9-11 & Conspiracy	19793	5190	2001	Unverifiable
8	MH370 Conspiracy	MH370 & Conspiracy	2014	660	2014	Unverifiable

4.3.3 Topic Flow Models and Visualization Views

The *Topic Flow* view visualizes the evolution of topics for a given rumor, employing a Sankey graph to represent how topics flow inside a rumor. In *Semantic Topic* view, for any two continuous submissions $S_1 = \{T_{11}, T_{12}, ..., T_{1m}\}$ and $S_2 = \{T_{21}, T_{22}, ..., T_{2n}\}$, a link from topic T_i in S_1 to topic T_j in S_2 is created the semantic similarity score between two topics T_i and T_j is above a threshold (default 0.0). The semantic score is computed using Google Tri-gram Method [90], which uses Google Web 1T datasets [21]. This affords the perception of topics being discussed when the rumor increased in attention and the degree to which they are semantically related. The *User Topic* view links two topics in the following way: for two continuous submissions $S_1 = \{U_1, T_1\}$ and $S_2 = \{U_2, T_2\}$ where $T_i = \{T_{i1}, T_{i2}, ..., T_{in}\}$ and $\mathrm{U}_i = \{U_{i1}, U_{i2}, ..., U_{in}\}$, there is a link from topic T_i in S_1 to topic T_j in S_2 if $U_1 \cap U_2$, the number of common users between two submissions S_1 and S_2, is above a threshold (default 1). The Sankey graph based on this connection allows the viewers to perceive the volume of migration of users from one topic to another along the rumor life cycle. Sankey graphs have been used before to visualize topic flows in OSNs [157], though with a construction based on influence models, instead of topic changes.

4.3.4 User Models and Visualizations

It is important for a complete rumor analysis that user activity is also understood. For that we model and visualize types of behavior towards spreading the rumor as well as the level of activity (see Objective 2).



(a)

(b)

Figure 4.6: User Flows in OSNs for three types of users. (a) “MH370 Conspiracy” (b) “Obama is a Muslim”

4.3.4.1 User Spreading Behavior Models and Visualization Views

The *User Spread* view reflects our formalization of the problem of how users interact with a rumor. RumorFlow adopted the rumor spread model theory proposed by Daley and Kendall [38]. In this model, N is the number of users that interact with this rumor. In the beginning, one user learns about this rumor from another source and tries to spread it in Reddit by posting a submission. Other users will read this submission and start spreading to other members on the website. N users will be categorized into one of the three categories: spreaders, ignorants, and stiflers, which are denoted as S, I, and R respectively at a given time. Spreaders are people who actively spread the rumor; Ignorants are people who are ignorant of the rumor; Stiflers are people who posted about the rumor, but are no longer interested in spreading it.

This model demonstrates rumor spread theory through a pair-wise contact between spreaders and other categories in the population. When two actors interact with each other in a rumor, one actor could infect the other in one of the three scenarios: $S+I \xrightarrow{\alpha} 2S$ when a spreader interacts with an ignorant, the ignorant will become a spreader at a rate α. $S+S \xrightarrow{\beta} S+R$ when two spreaders meet and one of them becomes a stifler at a rate β. $S+R \xrightarrow{\gamma} 2R$ when a spreader meets a stifler and the spreader becomes a stifler at a rate γ.

The rate α, β, and γ are calculated in the following procedure. RumorFlow adapted this rumor spread theory in Reddit as a stochastic process on $P = \{S, I, R\}^N$ where N is the total number of users for a rumor in a dataset. At a time t, the state of user i is $X_i(t)$. The procedure is as follows:

- The rumor starts when the first user A spreads this rumor by posting a submission ($S = 1, I =$ total number of users at time $t_0 - 1, R = 0$).
- All the users who have a comment on this submission are ignorants at t_0 and will become a spreader after.
- At time t:
  - If user j posts this submission, user j is a spreader.
  - If user j has a comment on this submission:
    - User j is a spreader if user j has a comment at $t - 1$.
    - User j is a stifler if user j has a comment at time $t - 2...t_0$.
    - User j is an ignorant if user j has no comments at time $t - 1...t_0$.
  - User j will become a spreader after time t.

Two distinctly diverse user flows are represented in Figure 4.6. The first "MH370 conspiracy", indicates a fluctuation of ignorant users, that is, those that participate for a short period, throughout the first stages of the rumors, and a larger number of stiflers over spreaders. For the second "Obama is a Muslim" the interest grows over time (that is, more ignorant users potentially getting into the conversation) and there are more spreaders than stiflers. Eventually, both user flows die out, but a few minor outbreaks occur which agrees with the rumor spread theory in Daley and Kendall [38]. From the visualizations, we see that the number of ignorant users is the highest, which is consistent for all rumors. Only a few users become active spreaders and stiflers.

4.3.4.2 User Interaction Models and Visualization Views

In RumorFlow, a Directed graph is assembled for a particular rumor having users as nodes and their connections as edges. An *User Interaction* view is from that graph as a node-link diagram. An edge connects from user U_i to user U_j if user U_i comments on a post or comments on a comment of user U_j. From this graph, we calculate network properties such as betweenness, closeness, and clustering centrality for each user, and code those by the size of the user node.

Betweenness: A user with a high betweenness centrality score means that this user can spread information to more other users in the network. This user plays an important role in the popularity of a rumor.

Closeness: A user with a high closeness score represents that information can traverse from this user to all other users faster. If a rumor is originally from these users, they have more chance to be spread further.

Clustering: Two users with a similar clustering coefficient score have a tendency to be in the same community.

The rationale is to locate opinion leaders and be able to observe their actions. For instance, a user with a high value of betweenness is key to make a rumor flow since he or she needs to have been being both commenting on other submissions and commented upon to achieve that centrality rank.

4.3.5 Topic/Word Cloud Views

Tag clouds have been used widely as an easy way to provide an overview of a corpus of text [140]. In RumorFlow, we propose and provide static context-specific tag clouds to make sense of the current discussion topic. The system has two tag clouds; a *Topic Cloud* view built from Wikipedia concepts for the rumors (top right of Figure 4.1), which changes from all rumors to a specific rumor if one is selected; and a *Word Cloud* view (bottom right of Figure 4.1), that reflects the actual words in a post and its comments when one is selected. Although comments are noisy, many words can call the attention of the analyst to content. Both *Word Cloud* view and *Topic Cloud* views discussed earlier, are linked to other views, showing posts and comments that carry that word or concept and adding a new dimension to the analysis.

4.3.6 Visualization Interactions and Other Functionalities

The RumorFlow prototype supports a robust set of interactions. Users can select a subset of rumor datasets that they are interested in. When a rumor is selected on the visualization, other rumors will become transparent in color. When a set of rumors is chosen from the *Dataset* options, the *Rumor Flow* view is reconstructed containing only the data sets of interest. Top and bottom visualizations can be amplified and panned. Each selection from the top view generates a visualization on the bottom view. Thus, if the top view is the *Rumor Flow*, selections of posts generate the visualization of the comment tree, selection of user sub-stream generates the *User Spread* view, and selection of the topic substream generates the *Semantic/User Flow* view. In particular, when user names are selected, a plot of the number of comments for that user in each of the main posts is shown in the bottom view (*User Comment Plot*). In this case, it is very clear to see the action of top or central users throughout the rumor's life cycle, for all rumors on screen. Users can be selected from the rumor flow, from the user graph, or by name, using a search window.

Additionally, a fisheye on the *User Interaction* view allows inspection of local details that might be hidden by clutter. Edge bundle [84] is available for the graph views to decrease global cluttering in the usually large user graph. Color palettes can also be changed to user's preference or need. The prototype allows analysts to recover all comments from a particular post through a link that appears on selecting a post. New data can be automatically retrieved for other rumors, though the pre-processing for a reasonably large rumor can be quite slow.

4.4 Results

This section will present a set of use cases and the results of the preliminary analysis by potential end users.

4.4.1 Implementation

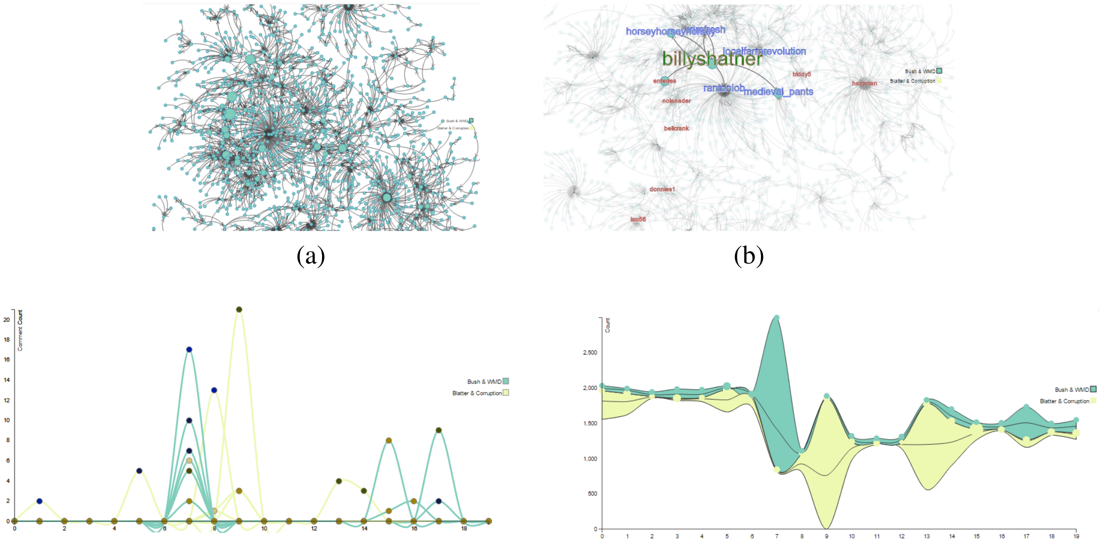
For data collection, RumorFlow adopts a service-oriented architecture approach to collect and visualize rumor data. All rumor-collected data are provided to the visualization system through JSON Restful web service from a JAVA backend. After users select a particular rumor, the system will use Reddit API and jReddit, a JAVA open source project, to extract submissions and comments about this selected rumor. All submissions that do not have any comments, "up" votes or "no" votes are removed, as we assume that users are not interested in them. Stop words are also removed.

RumorFlow adopts HTML5, JQUERY MOBILE, and D3 to display rumor data. JSNETWORKX is used to calculate node centralities. The statistics of the collected data is shown in Table 4.1.

4.4.2 Case Studies

4.4.2.1 Characteristics of a Rumor

We analyzed the rumor "Obama is a Muslim" and found some interesting patterns. For example, this rumor dates back to 2008, was debunked in 2009, but did not burst until 2012 which coincided with the US president election. This rumor still attracts a large number of comments from users in 2015. Using the *Comment Tree* view, we found that most users do not believe this rumor, but they want to make a joke about it. User "spaceghoti", for instance, has 70 comments in one submission. Many users comment on this user's comments and he/she also tries to defend this comment by replying to others. Most of these comments are very thorough and detailed. This is one of the reasons this user is most influential in



(a)

(b)

(c)

(d)

Figure 4.7: Analysis of user activity for rumors "Bush & WMD"(green) and "Blatter & Corruption"(yellow) (a) partial user graph (b) selected top centrality users (c) User Activity (as message counts) for selected users (d) Rumor Flow for both rumors

this rumor. On the other hand, user "spinninghead" is the most influential user in two other submissions about this rumor. He/she also has a few other comments in the rumor "Bush knew WMD" and "9-11 - Conspiracy". This is revealed by clicking on their names from either the *Rumor Flow* or the *User Interaction* views. In summary, for the "Obama is a Muslim" rumor, most users either joke or reject it with a detailed explanation; also, only a small portion of users actually approved of the rumor.

4.4.2.2 Rumors and their users

In this use case, we examine more relationships between user actions and rumor spread. We select "Bush & WMD" and "Blatter & Corruption" rumors, both having some relation to actual facts. We start by examining the *User Interaction* view for each of these rumors and selecting, from the graphs, the users with high betweenness centrality. In principle, these are influential users by commenting and being commented upon at important points in the conversation. The sequential steps in that analysis are shown in Figure 4.7. For both *User Interaction* graphs, we have selected the top ranked users based on the betweenness. One of the graphs before and after such selection are shown in Figure 4.7 (a) and (b). In Figure 4.7 (c) and (d) it is possible to observe that, for both rumors, the most central users act at similar time stamps and these times are on, or just before, the most active sessions of the rumor, suggesting that a few active users can account for the spread of the rumor in crucial moments.

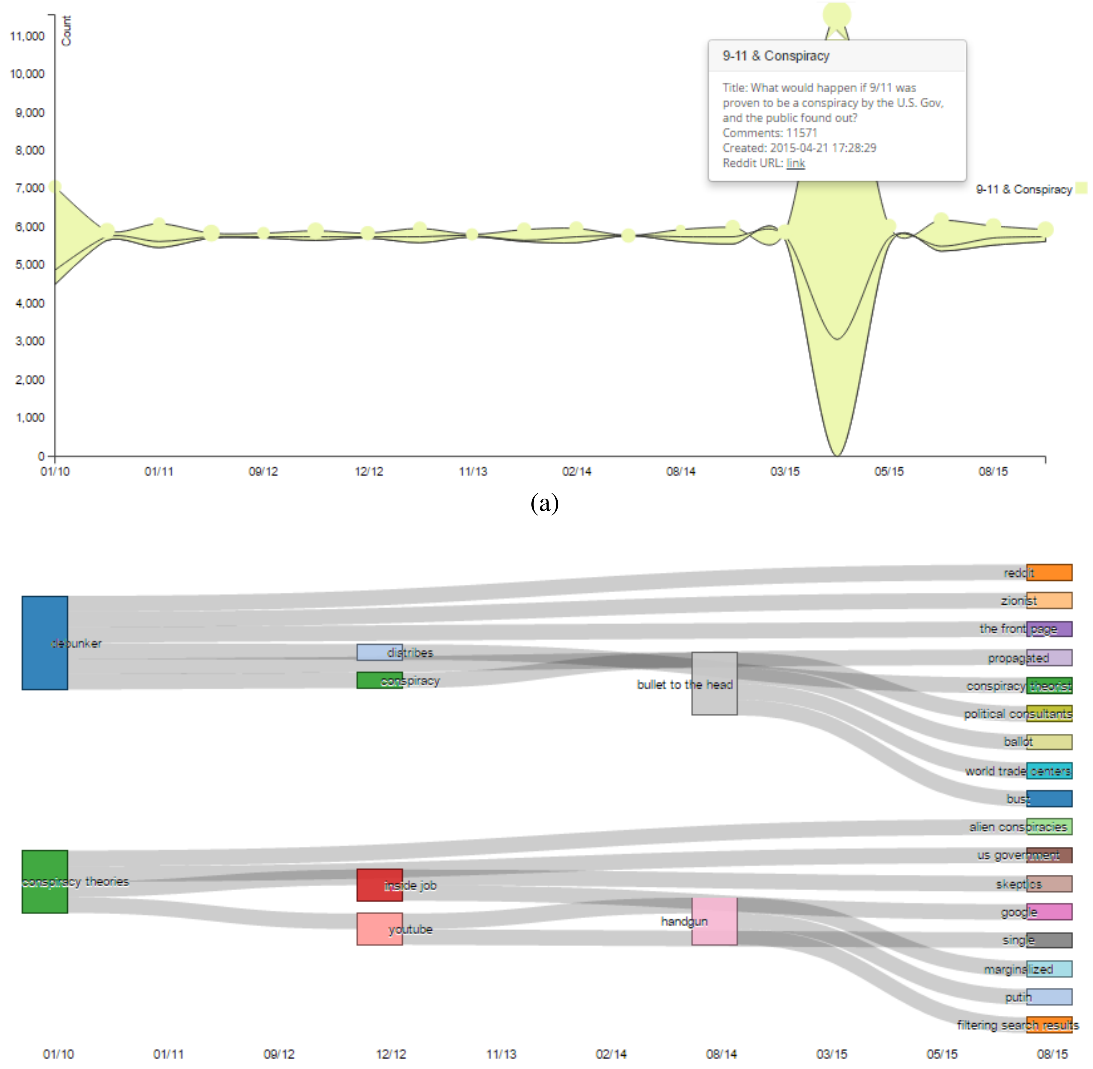
Another interesting observation afforded by the visualizations of these rumors is that for both, the most controversial posts (larger circles on time slot 15) are not necessarily the most commented on. Examining further, one of them has to do with "Jack Warner declarations against Blatter", and the controversy comes from the negative reaction of users to the perception that he was blaming others for his own actions. The second controversial post has to do with allegations that a CIA briefer had declared President Bush knew there where no weapons of mass destruction before going into war with Iraq. The negative trend has to do with most comments telling that as a lie and blaming the media for it.

4.4.2.3 The Big Rumor

From the rumor datasets collected thus far, the largest one in terms of both the number of users and of comments is the rumor "9-11 & Conspiracy". Examination of the rumor (Figure 4.8) shows that most of these elements occur as a response to a single post, which asks what would happen if 9-11 was a conspiracy and American citizens found out. It is also one of the most controversial posts of the whole set of rumors. Additionally, the user sub-stream reveals, and the user activity curve confirms, that comments do not come from repeat users as much as from different users. In fact, the most active user has only 23 comments for the larger post. The other popular posts, a debunker post with 2500 comments and a list of "reasons why it was a conspiracy" with 822 comments, have more active repeat users (190 and 85 comments respectively). The particularity of widespread participation with very few repeat players is confirmed by the *User Interaction* graph, showing extremely uniform centrality throughout, few users with large betweenness, low and very uniform closeness, and practically no community forming.

The user-connected *User Topic* view also reveals interesting patterns. First, there are two disjoint sets of topics along the rumors' life cycle. This means users that participate in one flow of topics are not the same users that participate in the other flow. The first flow starts with the debunker post, and the second one starts with conspiracy theories. The subject of conspiracy only occurs in the debunker flow later, as an opposition to the debunking. Meanwhile the conspiracy flow is fed by a number of different so-called "evidences". A second clear observation is the topic identified by the expressions "bullet to the head" in the first topic set and "hand gun" in the second topic set, that appear to be out of context.

By using the link feature between topic and stream, it is possible to find out that the post related to these subjects refers to an author of 9-11 conspiracy books having been killed with a single shot to the head.



(a)

(b)

Figure 4.8: Rumor Flow and Topic Flow (user) to the rumor “911 & Conspiracy” (a) a single post posing a question has more than 11k comments, followed by a debunker post (2,5K) and a list of the reasons why it was an inside job (814) (b) Topic progress for the Rumor, connected by common users. Notice that users either share one chain of topics or a second one. In the top chain, conspiracy ensues as a contradiction to the debunker

4.4.2.4 Additional Observations

From the analysis of the datasets, many other observations could be drawn. We have been noticed that there is a distinct difference between the behavior of users regarding rumors involving politicians and rumors of conspiracy theories. In the political arena, there are more users that act across different rumors (some in all observed political rumors), while different conspiracy theories have different sets of users. Additionally, more users in the political arena are likely to comment along different time periods in the life cycle of the rumors.

4.4.3 Analysts' Preliminary Opinions

Although we used an agile approach to build RumorFlow by having a weekly meeting for six months with four experts in visualization and social sciences to clarify the underlying rumor flow models and improve the visualization views. We also submitted the system to three other analysts for a preliminary analysis (see Objective 3). The first one has a comprehensive experience in natural language processing, language detections, and is familiar with social science theories. The second one has a strong experience in text mining and mathematical models. The third one is an expert in text mining and visualization techniques. All have a working knowledge of strategies to handle microblog textual information and visualizations. In the short analysis session, they were instructed on how to use the system, after which each analyst was asked what aspects of rumors they are interested in. Finally, they examined the available rumor data and expressed their opinions and observations in free text. Analyst One was interested in the fluctuating nature of rumors, so she chose the top-down approach. Analyst Two chose bottom-up approach to find user trolls and their related behaviors. Analyst Three focused on the visualization techniques and usability.

Top-down approach

Analyst One explored the system with a top-down approach. At first, she took a look at all the rumors in the *Rumor Flow* view, and she was asked to select a rumor that she is interested in. She moved the mouse over the rumor "HIV & Cancer" as this is the first time she heard this rumor and selected the stream to highlight it. When selecting *Semantic Topic* view, she found some interesting topics, such as "aches and pains" and "fevers" that are related to HIV symptoms. In addition, looking at the rumor, she found one submission that has hundreds of comments TIL Researchers have taken the HIV virus, modified it and then used it to reprogram cancer patients white blood cells to attack and completely kill off the cancer and analyzed its *Comment Tree*. When inspecting the rumor *User Spread* view, she found that the number of active spreaders is very low, but this number did not change throughout the rumor. After a further visual aggregation of users with high centrality score using tooltips and fisheye feature, she found this user has most comments in one submission and a few other comments in other submissions in the same rumor and other rumors. With various visual encodings of the *User Interaction* view, she found a disconnected cluster of users as shown in Figure 4.9. By hovering and expanding the central node, the analyst observed that this cluster seems to belong to all users about the submission which is about "Dexter Holland Ph.D. thesis about HIV & Cancer" and these all users only comment in this submission only. Here are her original comments: "I liked the feature of analyzing topics/words in comments: the word cloud, the ability to search the comments for a given word, being able to go down to a text of a comment in a flow or a chart — zooming back and forth from an overview to the detail of a particular comment. I also liked the analysis of the users: the analysis of the number of different types of users ('ignorants', 'spreaders', etc) - this seemed very interesting to me. I also find it interesting to be able to spot active / influential users and filter the comment text for particular users. The investigation of the number of comments and users in a relation to the time (the main view of the rumor) seemed very interesting for me.". She also suggested to connect *Semantic/User Topic* view with the *Rumor Flow* view which we have integrated her suggestion to the system.

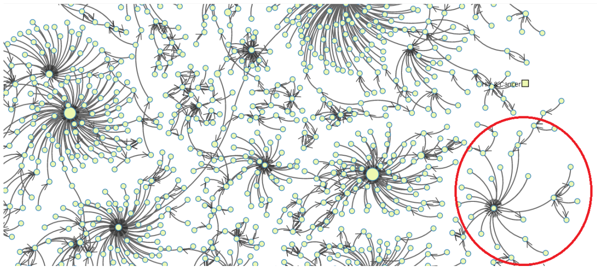


Figure 4.9: A disconnected cluster from User Interaction Graph. This cluster represents how a disconnected community discussing a specific topic inside a rumor.

Bottom-up approach

Analyst Two inspected RumorFlow using a bottom-up approach. We first asked the analyst to go to the *User Interaction* views and inspect these graphs. As he is interested in rumors about "George Bush" so he first took a list of users with a high centrality score in the rumor "Bush knew WMD". He knows from experience that users with high centrality scores may be trolls in a rumor. From the raw comments in *User Comment* view, he analyzed a list of user comments and found one user "hazzman" that wants to spread this rumor further. This analyst decided to start from exploring original rumors that this user troll involved in.

From the *User Comment Activity*, he found that this user involved in both the rumor "9-11 & Conspriracy" and "Bush & IQ". This user seems to be interested in topics about "George Bush" based on his comments and wants to spread rumors about this topic. Here is the original analyst comment "This rumor flow visualization tool could be useful to find some potential users and commentators who propagate rumors along with their motives. There are many powerful visual components in the tool that with different combinations could find some important insights. To successfully use this tool requires some training and this might hinder to gain its popularity among general users."

Interactive exploration

Analyst Three freely used the system without any guidelines. He is interested in visualization interactions and designs and is impressed with the number of interactions and visual analysis that he can explore. Here are some of his general comments: "In the Rumor Flow, show comments or users when zooming in." "The numbers of the comments and users should be shown on the stream.", "In the User Flow, Show if a user is repeated.", "In the Sankey Graph, some connected topics are not meaningful like connecting 'Obama is a muslim' with 'Obama'". "In addition: the Widgets should carry more meaningful information other than the names of the connected topics, such as the context they appear in.", and "Vertical words are harder to read. Word clouds are easier read with Horizontal word only." We have carefully integrated most of this analyst's comments into RumorFlow.

All three analysts were confident the proposal and the system have a potential to provide valuable insight into rumor spreading and user behavior towards it. Two of them were very excited about the system and the kinds of observations they can produce with it. Two analysts agree that the use of the system requires training and one mentioned that this could delay adoption by general users, mostly due to the large variety of functionalities. A few very specific criticisms on the understanding of some visualizations (e.g. peaks and valleys in the rumor flow and the interpretation of user flow) were mentioned. Interestingly, the techniques that one of the analysts had problem understanding were the precise ones most praised by the other two analysts who detailed their experience with the techniques. Due to the size of the system, we, of course, expected many improvement suggestions such as these. All analysts observations are being given due consideration for the follow-up of the system and planned targeted user studies.

4.5 Discussions and Conclusions

The RumorFlow approach has been demonstrated to allow a number of very important insights into rumor evolution, topic change and user activity in online social networks, that would not otherwise be available from the original textual form of the data. Additionally, a large number of associations between different rumor components (life cycle, topics, user importance and distinct user behaviors) can be observed and some current social science theories available can be confirmed in the context of online social networks.

The approach bridges a gap currently existing in the analysis of microblog data related to rumor dynamics, although the framework and corresponding system could also be used to analyze any type of submission-and-comment-based online communication, such as blogs, news posts, and discussion lists. The design, however, is meant to associate aspects that are typical of rumor spreading, such as simultaneous analysis of user and topic, rumor and topic and rumor and user. This differentiates the proposal from previous opinion or influence based visualizations. Social science research and analysis could benefit from the system and the approach, as mentioned previously in the text. It is our intention to create a layer of an application-specific interface to allow access by typical users in such human behavioral studies.

Investigative applications of various kinds can also benefit from the framework. For other applications and general users, we intend to build a set of training-by-example tools and video tutorials. All variations of the system will be publicly available, as well as all the data collected. Two particular important extensions are planned. One involves using the system to inspire and extract features for categorization of user behaviors towards rumors, and a second extension involves adding the possibility for users to remove and add information, influencing the layouts. Topics, for instance, are currently extracted with Wikipedia concepts, but users will be able to suggest topics, that can then be detected and added to the current visualizations. The system handles large amounts of data at once in near real time, except for the preprocessing phase. However, some networks reach tens of thousands of users and hundreds of thousands of comments, users and for those, we intend to build a parallel processing framework so as to analyze rumor in massive user networks.

Chapter 5

Toward Understanding How Users Respond to Rumours in Social Media

As the spread of rumours has been increasing every day in online social networks (OSNs), it is important to analyze and understand this phenomenon. Damage caused by the spread of rumours is difficult to handle without a full understanding of the dynamics behind it. One of the central steps of understanding rumour spread is to analyze who spreads rumours online, why, and how. In this research, we focus on the steps *who* and *why* by describing, implementing, and evaluating an approach that studies whether or not a group of users is actively involved in rumour discussions, and assesses rumour-spreading personality types in OSNs. We implement this general approach using Reddit data, and demonstrate its use by determining which users engage with a recurring rumour, and analyzing their comments using qualitative methods. We find that we can reliably classify users into one of three categories: (1) “Generally support a false rumour”, (2) “Generally refute a false rumour”, or (3) “Generally joke about a false rumour”. Combining text mining techniques, such as text classification, sentiment analysis, and social network analysis, we aim to identify and classify those rumour-spreading user categories automatically and provide a more holistic view of rumour spread in OSNs. The basis of this chapter is from the published paper [48].

5.1 Introduction

Online rumours, which are truth-unverifiable statements in online social networks (OSNs), are popularly spread in uncertain situations [137]. As billions of people and organizations are connecting with each other through social interactions, breaking news, and sports events, OSNs has become a popular source to share credible information [95]. As well as spreading credible information, OSNs can spread rumours [119, 31]. Problems like rumours going viral are not isolated and prompt the question of how to identify and limit the spread of rumours in OSNs. In an effort to constrain the spread of rumours, many researchers are trying to detect rumours [57, 121] and the original sources of rumours [141, 142]. However, little work has been done to understand who spreads rumours online and why.

People spread rumours for a variety of reasons. Bordia and DiFonzo [20] studied, from a psychological viewpoint, what motivates people to spread rumours in a social network. The authors identified three motivations for people to spread rumours: (1) fact-finding, (2) relationship-building, and (3) self-enhancement. People who are motivated by fact-finding are aiming to arrive at a valid and accurate understanding of rumours through a problem-solving process. In contrast, those motivated by relationship-building are simply interested in interacting with other people by sharing information about particular rumours. The same study pointed out that those with self-enhancement as motivation are either consciously or unconsciously simply spreading rumours. Researchers have tried to group users into the same group based on link predictions [106] and content characteristics [161]. In this work, we aim to automatically detect users based on how they interact with a rumour with regards to Bordia and DiFonzo's rumour-spreading theory.

As people have a tendency to believe misinformation, which are false rumours, and misinformation is more likely to be spread [101], we focus on studying whether users spreading misinformation in Reddit can be divided into one of the three categories derived from Bordia and DiFonzo's rumour-spread motivation theories: (1) "Generally support a false rumour" (self-enhancement), (2) "Generally refute a false rumour" (fact-finding), (3) or "Generally joke about a false rumour" (relationship-building). To achieve this goal, the proposed research collects rumour-related data from Reddit and applies text mining techniques and social network analysis to analyze and visualize users in those three categories.

To date, most of the work in this emerging area has been conducted to: detect rumours, limit the spread of rumours, and identify the source of rumours. However, in order to develop effective methods for rumour detection and prevention in OSNs, we first need to understand who spreads rumours online and why. This motivates us to propose the following research statements:

- Based on user activities in Reddit, could we determine if there is a specific group of users that is greatly interested in discussing and spreading rumours?
- Based on user activities in Reddit, could we determine if there is a rumour-spreading personality type in Reddit who, for example, "Generally supports a false rumour" (SUPPORT), "Generally refutes a false rumour" (REFUTE), or "Generally jokes

about a false rumour" (JOKE)?

- Will visualizing rumour spread in Reddit provide better insight into how users interact with rumours?

This chapter makes the following contributions:

- Collecting and analysing user posting behaviours in Reddit about a specific rumour. Based on users' interaction, determine if there is a group of users that is actively spreading rumours.
- Using social network analysis, visual analytics, content analysis, and text mining techniques, the system classifies the active rumour-spreading users Bordia and DiFonzo rumour-spreading theory [20] into one of the three categories: (1) SUPPORT, (2) REFUTE, and (3) JOKE.
- The experimental results using text mining techniques confirm and support our approach.

This chapter has the following structure: Section II reviews related work, Section III describes how we collect the data, Section IV describes the methodology, and Section V analyzes and discusses the results.

5.2 Related Work

Previous work in this area is concentrated in three main areas: mining online social networks, rumour analysis, and visualizing rumour spread in OSNs. It is important to highlight that the research focuses on rumour spreading in social media. It is implemented for Reddit data, and illustrated with the "Obama is a Muslim" rumour.

5.2.1 Mining Online Social Networks

Modern OSNs produce vast amounts of user-generated content. Analyzing content at this scale requires algorithmic support, typically in the form of data mining. Falkowski et al. [59] used statistical analysis and visualizing OSNs to study the dynamics and evolution of subgroups in communities. The authors proposed different community similarity measures and grouped similar communities into the same cluster, and later visualized community-clustering results to analyze their dynamics and evolution. The experiments showed that this method could detect the fluctuating nature of an online community. Liben-Nowell and Kleinberg [104] adopted knowledge from social network analysis (e.g., centrality features), graph theory (e.g., graph distance), and social sciences to gauge the effectiveness of network-proximity measures. Based on these measures, the authors tried to predict new interactions that would have a high probability of occurrence in the near future. Golbeck et al. [25] predicted the personality type of a user based on the user's Facebook profile. Dang et al. [40] uses text syntactic and semantic similarity to map related Tweets to users' profiles.

None of the studies examine Reddit data; Reddit is under-studied in the social media research literature, despite being one of the most-visited social websites in the US.

5.2.2 Rumour Analysis in Online Social Networks

Within the field of social media research, there has been previous work focused on rumour analysis, using a variety of approaches (including data mining). In this research, we are only interested in using rumour-related memes to pinpoint which users are spreading or refuting rumours in OSNs.

The modern study of rumours dates back to 1944, in the work of Festinger et al. [63]. The authors studied the origin and spread of rumour in a specific neighbourhood community by intentionally starting rumours. After six months, intensive open-ended interviews with the residents in this neighbourhood about the rumours were recorded. The experiments found that not everyone who heard the rumour spread it further, and existing friendship connections between people increased the probability of the rumours being spread. Due to the intrinsic long-lasting nature of rumours and the difficulty in collecting rumour data, rumour analysis research experienced a lengthy hiatus until the popularity of OSNs in the 2000s.

In most OSNs, information is disseminated and stored permanently, so researchers are able to use the data to study rumours and their analysis more effectively. Marett and Joshi [113] investigated underlying motivations for posters and lurkers spreading information and rumours in a local online community. Posters are users that regularly post their experiences and stories in OSNs, while lurkers are users who only read the posts from other posters. The authors gathered posting data from a local university forum and conducted an online survey for both posters and lurkers in that community to understand why they spread rumours. The results showed that the intrinsic motivation, i.e., "the doing of an activity for its inherent satisfaction rather than for some separable consequence" [139], played a critical role in motivating posters to share information and rumours in this online community. One limitation of this approach is that it relied on self-reported responses from users to hypothesize why users spread rumours.

Recently, researchers have focused on using machine learning and the availability of big data in OSNs to study the spread of rumours and detect them automatically. Shah and Zaman [142] built a probabilistic model graph based on network structure and rumour-infected users. This model provides a rumour centrality score for each node in the graph, and the node with the highest rumour centrality score is the source of rumours. Qazvinian et al. [132] proposed a general supervised-learning framework to identify rumours in Twitter. Retrieved tweets were manually labeled as either being related to rumours or not. Based on this training set, the machine-learning framework classifies whether or not incoming tweets are about the rumours.

Although researchers have achieved some degree of success in detecting rumours and understanding their pattern, little work has been done to investigate who spreads rumours in OSNs and why. The closest work to our research is that of Buntain and Golbeck [25], who presented an automated method for identifying the "answer-person" role in Reddit based on user interactions. Users filling this role only respond to questions by other users and do not get involved (or have only limited involvement) in other discussions. They first manually analyzed data collected from Reddit to determine if this role exists. Next, they designed a feature set that characterizes this role and uses this feature set to classify more answer-person roles in the network. Our goal parallels the work of Buntain and Golbeck; our objective is instead to determine if rumour-spreading users exist in OSNs. To the best of our knowledge, no similar work has been done on studying rumour-spreading users in Reddit.

5.2.3 Visualizing Rumours in Online Social Networks

Table 5.1: Examples of submissions about the rumour "Obama is a Muslim".

No.	Title	Date	No. Comments
1	People in Middle America believe that Obama is a Muslim	2007	234
2	Is Obama a Muslim? About.com poll: 57% Yes, 37% No, 10% Undecided. Let's correct this.	2008	299
3	Do you think Mr. Obama is a Muslim or a Christian? ....I know, I know...	2009	28
4	Scientist asks why Americans believe Obama is a Muslim	2010	279
5	Iowa GOP Focus Group: Obama Is A Muslim	2011	169
6	Do you think Barack Obama is a muslim? Alabama Republicans: 45% say yes. Mississippi: 52%	2012	169
7	My Orthodox rabbi says President Obama is halachicly a Muslim...	2013	41
8	Proof that Obama is a Muslim!!!	2014	30
9	Poll: 54% of Republicans say that, "deep down," Obama is a Muslim	2015	2923

One of the most effective ways to study rumours in OSNs is to visualize the paths and patterns of the spread. Some recent scholarly and industry-led projects relied on visualizations to show how online rumours are spread. Ratkiewics et al. [135] developed *Truthy*, a supervised-learning visualization framework, to identify misleading political campaigns by collecting, analyzing, and visualizing messages through the Twitter network. First, this framework detects any emerging memes which are a unit of information that can be spread from users to users in Twitter. Next, content, network and sentiment analysis are used to classify whether a meme is rumour-related. Finally, the path and pattern of rumours are visualized for further research. Similarly, The Guardian [39] visualized how rumours identified by reporters covering the story about the 2011 UK riots spread on Twitter by grouping related Twitter messages into the same cluster. Dang et al. [45] proposed RumourFlow, a visual analytics framework, which allows analysts to collect, analyze, and visualize rumour spread in Reddit by exploiting the use of social science theories, text mining techniques, information diffusion models, and sentiment analysis. In this work, we use visualizations, text mining techniques, and social network analysis to analyze and understand how rumour-spreading users interact with rumours and with other users.

5.3 Reddit Social Network

Reddit, which claims to be the front page of the Internet, is a social news website where users can actively participate in content creation. Registered users discuss a wide range of topics such as politics and world news every day. User-submitted content, called *submissions*, can be text content and direct links to other online content. Redditors can comment or vote (up-vote or down-vote) on each submission; these interactions determine the rank of the submission on the site. Redditors organize content into subcategories called subreddits. Every Reddit submission has the following elements:

- **Title:** The title summarizes the topic of the submission and is usually very short and concise.
- **Comments:** for each submission, users can post a comment that expresses their opinion about the submission; comments are organized hierarchically, so users can post comments on other comments. Users can also vote the comment up or down.
- **URL:** each submission may contain a link to an external source of information that is relevant to the submission.
- **Image:** each submission may also contain a link to an image to illustrate what the

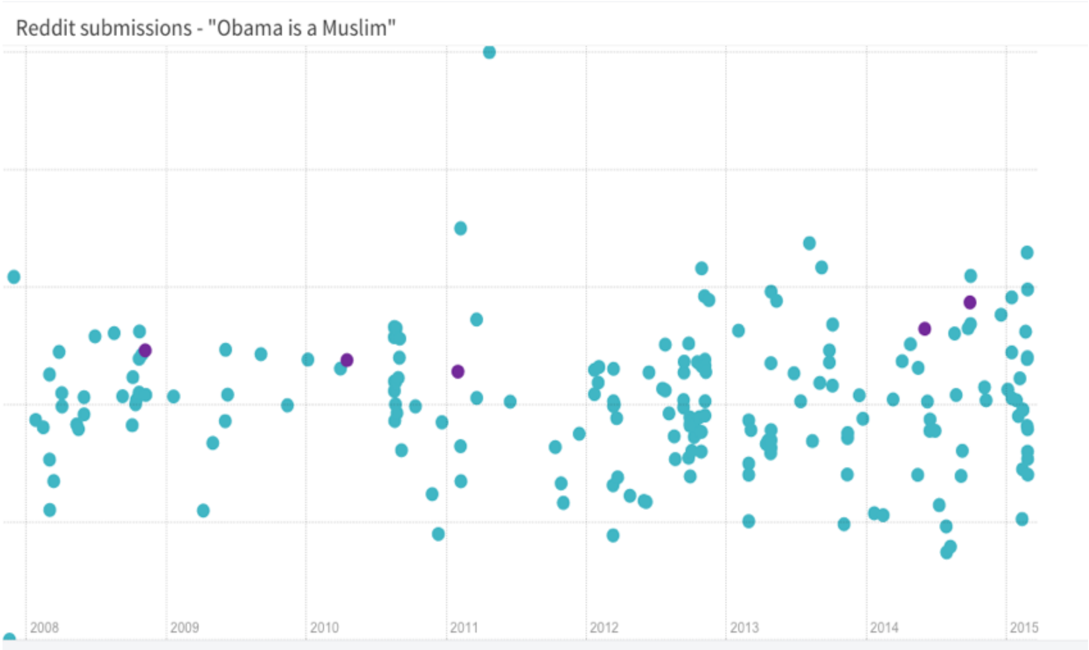


Figure 5.1: Submissions regarding “Obama is a Muslim” over time. Each node represents a submission, and the nodes are visualized in ascending order of posted time while the y-axis represents the number of comments of each submission. The purple nodes represent the submissions that the user “kickstand” has commented on.

submission is about; a thumbnail is displayed on Reddit.

Reddit.com is ranked as one of the most visited sites globally. The massive amount of data disseminated through Reddit every day makes it an excellent tool for analyzing and detecting rumour-spreading users in social media. Although Twitter has been the most popular source for studying rumour spread in OSNs [132, 135], Reddit has made a few inroads into the world of analyzing rumour-related memes [41, 25]. While Twitter and Reddit do share some commonalities, they are different in important ways. Twitter primarily circulates news through known cycles (e.g., “follow” connections), whereas Reddit promotes a constant stream of new links to all users through a simple bookmarking interface. This makes Reddit an effective source for studying the spread of rumour-related memes in OSNs.

5.4 Methodology

We describe our general approach to collecting, visualizing, and analyzing rumour-related data using Reddit as a specific implementation example.

5.4.1 Data Collection

To study the spread of rumours in Reddit, we need the following elements:

- A rumour.
- The truth about this rumour.
- Posts about this rumour.

We adopt Snopes.com as a reliable source for collecting, confirming or disapproving rumours. Snopes is a website that collects memes, urban legends, and stories with unknown or uncertain origins. It provides a wide range of rumours, from politics, altered images, to real photos with fake stories, and even hoaxes. Each rumour is categorized as true, false, partly true, multiple truth-values, unverifiable, or legend. The editors of this website verify and provide evidence that could be used for debunking or confirming rumours. We also collect submissions and comments that are related to a specific rumour being discussed in Reddit. There may be one or more submissions for each specific rumour, so we have to create a generic query to capture all of the rumour-related submissions. Since no repository for a Reddit rumour dataset exists, we use the Reddit API and jReddit (a JAVA open-source project) to extract submissions, comments, and other data views, such as image or URL content, about a specific rumour using predefined regular expressions.

We used the rumour “Obama is a Muslim” in Reddit from 2007 to 2015 as our case study due to its persistence, popularity, and controversy. We automatically searched the keywords “Obama & Muslim” from Reddit and collected 195 submissions, 26,421 comments from 11,125 users, 85 submissions containing a URL, and 29 submissions containing an image. As our primary interest was in users that are actively involved in rumour spread, we removed users that engaged in fewer than 10 comments in these 195 submissions. This reduced the number of users to 163, and is the dataset used most frequently in this chapter (note that we choose 10 as our threshold value to achieve a good representative sample of active users and to ensure statistical significance). Given our interest in not just assessing the existence of this group, but also in assessing if long-term participants in the conversation can be categorized into categories based on Bordia and DiFonzo’s described motivations. Two judges review comments of each user based on their repeated comment patterns and categorize them into one of the three categories SUPPORT, REFUTE, and JOKE (with Kappa agreement score = 0.85).

5.4.2 Data Visualization

For data visualization, this chapter uses RumourFlow [40], a service-oriented visualization framework to collect and visualize rumour spread. All collected rumour data are

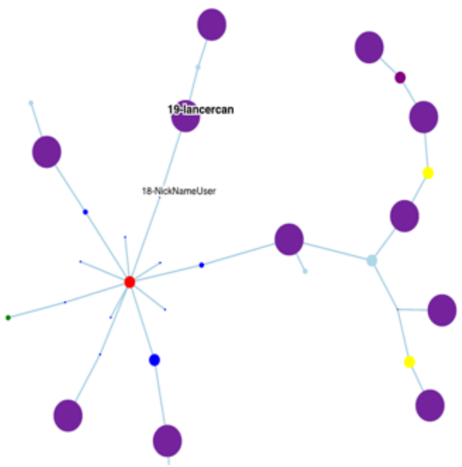


Figure 5.2: User “lancercan” interaction graph about the rumour. A node represents a comment for a specific submission, and a node size displays how many times a user comments on a specific submission. The red node represents the original submission, and comments in the same level are nodes of the same color (e.g, the blue nodes represent comments to the red nodes). The purple nodes represent the comments of user “lancercan” for this submission, and suggest “lancercan” is a frequent participant in this discussion.

provided to the visualization system through a JSON restful web service from a JAVA backend. For visualization, we adopt D3 and jQUERY to display rumour spread through a web-based application. The goals of this visualization framework are to provide a visual analytics tool for researchers and end users to explore different aspects of rumour spread in OSNs. It has two main views. The first view presents an overview of how rumours evolved over time as shown in Figure 5.1, and the secondary view describes how users interact with each submission about rumours by an egocentric network as shown in Figure 5.2. This framework also offers users easy access to search for a specific rumour in Reddit with their own keywords or for a specific user that comments about a rumour.

5.4.3 Approach to Analysis

After collecting the data, we adopted social network analysis, content analysis and text mining techniques, to analyze and visualize these contents.

Social network analysis (SNA) refers to the use of network theory for understanding social network data. Social networks have been widely used since the early twentieth century to depict a certain community and how people in this community interact [126]. Because of the analogy between online social networks and the structure of social hierarchy and stratification, the study and analysis of social networks has played an important role in understanding how OSNs work. We focus on how users interact with other users about this rumour using our visualization tool to explore the data, particularly connections between users and a longitudinal assessment of the prevalence and re-appearance of the rumour over the 9 years in our dataset.

Content analysis is a qualitative method that examines the meaning of textual data manually to identify and assess themes and patterns. We focus on the characteristics and content quality of user posts in each category of user (SUPPORT, REFUTE, JOKE) and manually review each comment in all three categories to identify typical patterns and themes.

Finally, text mining techniques, such as data classification, visualization, and sentiment analysis are used to validate if the characteristics of each user group found from social network analysis and content analysis could be classified automatically.

All rumours have a beginning and an end. A rumour may be considered true at one point but is debunked as false at a later point. As a result, we try to capture all submissions about a rumour and visualize its evolution from its start to its end so that end users can discover all facets of a rumour life cycle. An example of each submission about the rumour “Obama is a Muslim” in each year from 2007 to 2015 is shown in Table 5.1. An interesting observation is that the submission in 2015 still receives numerous comments from users. This suggests that the “Obama is a Muslim” rumour is still popular, even though it was first started in 2007.

5.5 Results

This section describes the analysis of the data, focusing on the highlights of the examination of the data using the visualization tool towards answering the research questions.

5.5.1 Rumour-discussing Users

Table 5.2: Rumour-spreading users about the “Obama is a Muslim” rumour.

Rumour-spreading Users	User Count	Percentage
SUPPORT	8	4.9%
REFUTE	41	25.2%
JOKE	85	52.1%
OTHERS	33	20.2%

First of all, we aim to determine if there is a group of users actively involved in rumour

Table 5.3: Examples of user comments in each category.

Category	Comments
SUPPORT	“He is a Muslim clearly.”
REFUTE	“This is actually a good point. The radical conservative movement doesn’t use language like the rest of the people. They don’t say what they mean, or what they think is true. They say things to achieve the desired result. So, if they think saying Obama is a Muslim will damage him, by all means they will say that. They use ”words that work”?”
JOKE	“?Eh you should come to the south and meet the people I have. Many people seriously believe he’s Muslim. Many people also think men have less ribs than women despite that we know 100% it’s not true. People are stupid.”

discussion and spread. Of the 11,000 users that have comments in the 195 submissions, 163 users have repeatedly interacted with one or more submissions by having 10 comments or more in those submissions. For example, how the user “lancercan” actively interacts with a submission about the topic “Obama is a Muslim” is shown in Figure 5.2. Another example shows how the user “kickstand” interacts with the submissions in the collected dataset in Figure 5.1. The use of stream and circles for visualizing time series graph has been used widely in the literature [165, 155]. This visualization helps to discover how a user is actively involved in discussing and spreading rumours. It shows that this user has repeatedly commented on this rumour since 2007 until 2015 in various submissions and years. These examples illustrate the larger group of users on Reddit that is very interested in discussing this topic for an extended period; the existence of this group is clear from the

Table 5.4: Examples of comment interactions between user groups.

Category	SUPPORT	REFUTE	JOKE
SUPPORT	N/A	N/A	N/A
REFUTE	**Rumour:** “Poll: 54% of Republicans say that, ‘deep down’, Obama is a Muslim”. **Comment:** “Dam it to hell, I knew he was a Muslim!”. **Response:** “I knew you are wrong.”	**Rumour:** “Poll: 54% of Republicans say that, ‘deep down’, Obama is a Muslim” **Comment:** “Funny, because I suspect if he were a closeted anything, it’d be a closeted atheist.” **Response**: “He’s an atheist because he don’t believe in god?”	**Rumour:** “Poll: 54% of Republicans say that, ‘deep down’, Obama is a Muslim” **Comment:** “Deep down, 54% of Republicans are idiots” **Response:** “I agree”
JOKE	**Rumour:** “Poll: 54% of Republicans say that, ‘deep down’, Obama is a Muslim”. **Comment:** “He’s clearly trying too hard to not look like a Muslim. That makes it obvious that he is actually a Muslim”. **Response:** “Except, he will always look like a Muslim”	**Rumour:** “Poll: 54% of Republicans say that, ‘deep down’, Obama is a Muslim” **Comment:** “Deep down, 54% of Republicans are idiots.” **Response**: “More proof that American voters have little or no memory.”	**Rumour:** “Poll finds 23% of Texans think Obama is Muslim” **Comment:** “Poll finds 23% of Texans are idiots” **Response:** “I like to look at the positive side: 77% are not stupid”

Table 5.5: Interactions between user groups; each row represents how frequently users in that category reply to comments or submissions in each of the three categories.

Number of connections	SUPPORT	REFUTE	JOKE
SUPPORT	N/A	N/A	N/A
REFUTE	25	5	4
JOKE	63	49	34

data.

A breakdown of the user-category dataset statistics is shown in Table 5.2. The data demonstrates that most users either joked about this rumour or refuted it with a detailed explanation. Only a small portion of users supported this rumour. An example of user comments in each category is shown in Table 5.3. Users in “OTHERS” categories seems to discuss related points with the rumour. Some of them discuss religion related topics.

5.5.2 Cross-Category User Interactions

We also counted the possible connections among users in the three categories (i.e., who replies to whom) to explore how users in different categories interact with each other. An example of each interaction between user groups is shown in Table 5.4 and detailed statistics on how users in one group interact with users in another group are shown in Table 5.5. In these two tables, each row represents how users in that category reply to users in the other three categories. There is not enough data about how users in the category SUPPORT interact with each other or with users in the other categories. It is clear that users in JOKE category tended to receive more responses from other users. Users in REFUTE and JOKE categories share many interactions between them. One possible reason for this finding is that Snopes debunked this false rumour in 2009, so people are inclined to refute or joke about it; there were very few supporters in our dataset. Furthermore, users in JOKE are more likely to respond to a comment of a user in SUPPORT or REFUTE and are also more likely to have connections to users in the same category. This may suggest that humour may play a significant role in why this rumour is so popular.

We also investigated how users in all three categories interacted with a specific submission about this rumour. We found that submissions that posted a link or an image to an external source perceived as reliable received much more attention and many more comments from users than a submission without a link or an image. This suggests that rumours with an image or a link from external sources perceived as reliable are more likely to be spread further [64].

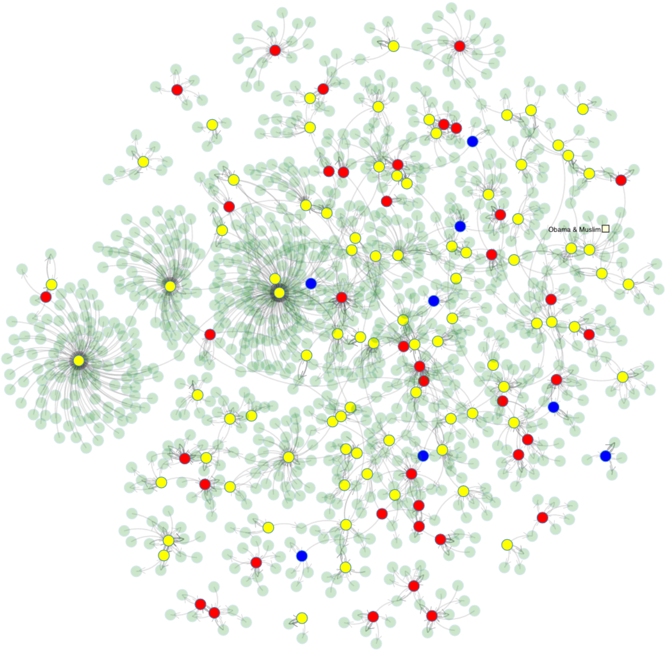


Figure 5.3: An example of "who replies to whom" ground-truth user graph. Yellow nodes: users in JOKE category. Red nodes: users in REFUTE category. Blue nodes: users in SUPPORT category. Green nodes: users that have no more than 10 comments but has a connection to the users in one of the three categories.

5.5.3 Content Analysis

Beyond understanding the users and their interactions, we also sought to analyze the textual content of submissions in each category. As most users that are actively engaged in this rumour do not believe it is true, we revisited the original dataset, which includes users who commented fewer than 10 times. We found that users in SUPPORT usually posted only one or two comments about this rumour. All of these comments were usually very short and had no back-up evidence or explanation. Here are a few examples:

"He's the kind of Muslim who?"
"I think he's a Muslim too"
"He's clearly trying too hard to not look like a Muslim. That makes it obvious that he is actually a Muslim."

For users in the REFUTE category, many comments were very thoughtful and provided in-depth explanations. Here are a few examples:

"Right, a politician would never lie or dissemble. If Obama says it, it must be true. I don't think Obama is a covert Muslim, but I wouldn't be surprised to learn that at some point in his life he was saying the [Shahadah.](http://en.wikipedia.org/wiki/Shahadah) His father was a Muslim before being an atheist. His mother ran off with a man to Indonesia and brought Barack Hussein with to the most populous Muslim nation. Barack doesn't strike me as a Muslim, but he may have real Muslim sympathies and may well have been exposed to Muslim indoctrination in his youth. Rejecting this possibility on the word of a lawyer and politician is your prerogative, but I prefer rational skepticism when it comes to politics."

"I don't care about any candidates religion unless they point it out as one of there qualifications for being elected to office. I can't remember Obama doing that except to deflect comments that he is a Muslim. Many republicans point out there adamant belief in Christianity and the belief that man was created by God, as stated in the First Book of The Bible: Genesis as a scientific fundamental. I cannot bring myself to vote for that type on nonsense. So I usually just waste my vote on a third party candidate."

Users in the category JOKE usually made a sarcastic comment or joke to refute this rumour. Here are a few examples:

"Mitt Romney's Birth Certificate. His Father was born in Mexico. Romney is just as 'foreign' as Obama is Kenyan or Muslim."

"Instead of convincing all those people they are wrong, we should just get Obama to convert to Islam."

In this instance, people are more prone to make a joke about it or refute it with a detailed explanation. Only a few people believe or support it.

5.5.4 Sentiment Analysis

In an online conversation, users' sentiment analysis has played a major role how this conversation becomes popular and its topic evolution [49, 163]. Each comment was parsed into sentences and each sentence is assigned a sentiment score: "positive","negative", and "neutral" using OpenNLP. We apply the concepts of sentiment polarity and subjectivity of Zhang and Skiena [163] for each user category in our ground-truth dataset as follows:

$polarity\_score = (p - n)/(p + n)$

$subjectivity\_score = (n + p)/N$

where p is the number of positive statements, n is the number of negative statements, and N is the total number of statements (including neutral statements). Polarity score represents if a user category is associated with the entity positive or negative, while subjectivity score depicts how much sentiment a user category garner. The polarity and subjectivity scores for each user category are shown in Table 5.6. REFUTE users have the highest polarity and subjectivity scores, while SUPPORT users have the lowest polarity and subjectivity scores. This can be explained as this rumour was debunked by Snopes in 2009 as a false rumour.

Table 5.6: Polarity and subjectivity score of each user category.

Category	Polarity	Subjectivity
SUPPORT	0.484	0.680
REFUTE	0.747	0.753
JOKE	0.638	0.705

5.5.5 Classifying Rumour-spreading Users

Using the content analysis, we observe that content characteristics in each rumour-spreading user group has its own characteristics. As a result, in this section, we explore if we could determine the user rumour-posting behavior automatically based on its content. For each user that has more than 10 comments, we transform them using the TF-IDF vectors, which reflect how important a word is in a document or a corpus (stop words are removed). Each user is represented by a vector:

User = $\{T_1, T_2, ..., T_n\}$ where T_i is TF-IDF score of term i by the formula $tf - idf_{t,d}$:

$$tf - idf_{t,d} \; = \; tf_{t,d} \times idf_t \; = \; tf_{t,d} \times log(\frac{N}{df_t})$$

where t is the term, d is the comment that has term i, and N is the number of users (documents).

After transforming each user comment data into a TF-IDF vector, we apply NaivesBayes classifier to those vectors and classify each user into one of the three groups: SUPPORT, REFUTE, and JOKE. Through various parameter settings, we achieve the best result with 80% accuracy using 10 fold cross validation and the dimension of the vector is 200. The classification result agrees with the manually classified data based on the two human assessors and further supports our hypothesis about the intrinsic content characteristics of each user group.

5.6 Discussion

The first research question asked if we could determine if there is a specific group of users that is greatly interested in discussing and spreading rumours based on user activities in Reddit. Our visualization tool allowed us to identify these users, and manual analysis of users in this highly-engaged category revealed persistent interaction with the rumour over 10 years for 163 users. As noted in our threats to validity section, it is possible to disguise high levels of interest in a particular rumour, but our approach is effective in many cases.

The second question asked if we could assess user activities and assign types to users based on whether they support, refute, or joke about the rumour. Our manual assessment of all user comments revealed that users did engage with the rumour in a consistent pattern. These users can be categorized into three groups: (1) “Generally support a false rumour”, (2) “Generally refute a false rumour”, and (3) “Generally joke about a false rumour”. We further examined how users in these categories interacted, and found that joking users were the most active. Using both social network analysis and content analysis provided us with some interesting results. Users in the JOKE category seemed to be the most active group that interacted with the rumour among themselves and with users in other categories. The content of user comments in REFUTE was explanatory and fact-driven, while the content of user comments in SUPPORT lacked details and evidence. Finally, applying text mining techniques allows us to identify those users automatically.

The third question asked if visualizing the spread of a rumour could provide better insight into how users interact with rumours. The assessment of this is subjective and qualitative, but we certainly found that a visual depiction of how rumour-spreading users interacted with submissions, with comments, and with other users was helpful in tracing and understanding the spread of the rumour. The longitudinal analysis showed a persistent rumour, and identified submissions on the topic that deviated from others based on an automated semantic assessment. The visualization tool allows both a high-level view and a detailed breakdown; some visualizations are presented in this chapter and are certainly helpful in drawing hypotheses that could not be driven without visual observation of raw data.

Threats to Validity

Our evaluation is based on a case study and our own observations about our method, both threats to external validity. Our results will require further validation before we can confidently assert that they apply generally.

Redditors can, and do, change their usernames, create new identities, or update/delete their own comments or submissions. Our counts of users engaging with this rumour is therefore a floor, not a ceiling or a precise measurement.

In this case study, we focus on a false rumour that was debunked by Snopes. We would also like to investigate if the proposed approach is still valid for rumours that are only partly false.

5.7 Conclusions

In this chapter, we presented a study about how users interact with rumours in Reddit. The results have shown that a specific group of users actively interacted with the chosen rumour. These users are categorized into three groups: (1) “Generally support a false rumour”, (2) “generally refute a false rumour”, and (3) “Generally joke about a false rumour”.

The use of social network analysis, content analysis, visualizations, sentiment analysis, and text classifications validate and support the proposed approach. Users in the category “Generally joke about a false rumour” seemed to be the most active group that interacted with the rumour among themselves and with users in other categories. The content of user comments in “Generally refute a false rumour” was explanatory and fact-driven, while the content of user comments in “Generally support a false rumour” lacked details and evidence. Finally, those users are grouped in one of the three categories automatically using text classification.

We illustrate our general approach using data from Reddit. This approach is also suitable for other OSNs (like Flickr or Twitter); however, OSNs do not always exhibit the same user behavior, so the specific results of our analysis are not necessarily true of other OSNs. Additional studies will be required to assess user behavior on other OSNs.

Chapter 6

Early Detection of Rumor Veracity in Social Media

Rumor spread has become a significant issue in online social networks (OSNs). To mitigate and limit the spread of rumors and its detrimental effects, analyzing, detecting and better understanding rumor dynamics is required. One of the critical steps of studying rumor spread is to identify the level of the rumor truthfulness in its early stage. Understanding and identifying the level of rumor truthfulness helps prevent its viral spread and minimizes the damage a rumor may cause. In this research, we aim to debunk rumors by analyzing, visualizing, and classifying the level of rumor truthfulness from a large number of users that actively engage in rumor spread. First, we create a dataset of rumors that belong to one of five categories: “False”, “Mostly False”, “True”, “Mostly True”, and “Half True”. This dataset provides intrinsic characteristics of a rumor: topics, user’s sentiment, network structural and content features. Second, we analyze and visualize the characteristics of each rumor category to better understand its features. Third, using theories from social science and psychology, we build a feature set to classify those rumors and identify their truthfulness. The evaluation results on our new dataset show that the approach could effectively detect the truth of rumors as early as seven days. The basis of this chapter is from the published paper [46].

6.1 Introduction

Online rumors are truth-unverifiable statements or news that are spread and discussed in Online Social Networks (OSNs). They commonly appear and are propagated in uncertain situations [137]. Recently, social media has been used as a means to transmit information, such as breaking news, sport events, and political statements [95]. Although social media provides a reliable way to spread information to a large population in a short time, it also has a critical drawback. For example, a lot of information in social media could be rumors that are spread maliciously. Rumors that are “False”, “Mostly False”, or “Half True” could cause a tremendously adverse effect on people’s lives. This raises the questions of how to identify, validate, and debunk the truthfulness of rumors. Researchers have tried to: detect rumors [57, 121], detect the original sources of rumors [142], identify who spread rumors [48], and rumor stance classification [108]. However, little work has been done to debunk and validate the level of the rumor truthfulness.

Studying rumor spread is an inherently interdisciplinary field. Social scientists have studied the intrinsic characteristics of rumor spread since the 1940s [63] and proposed theories on how rumors are propagated. For example, “The Basic Law of Rumor” [35] stated that the popularity of a rumor depends on the importance of its topic and the verifiability of its truthfulness. Recently, with advances in visualization and machine learning, it has become possible to apply knowledge from social science and psychology to better understand rumor spread in OSNs. Researchers have focused on the rumor veracity task [52, 96], i.e., given a rumor in social media and its related posts, to determine the veracity of this rumor (as “true”, “false”, or “unverified”). Most of the existing approaches tackle the problem from a machine learning point of view (e.g., trying various features and deciding what features produce the best result). These approaches may not be able to capture the changing characteristics of rumor spread [96]. In this chapter, the feature set of the rumor veracity task is derived from established social science theories. This not only provides more credible results but also explains whether social science theories could be applied to social media data.

Researchers have confirmed that false rumors, hoaxes, or fake news (another form of false rumors) are more prone to be spread further [115]. Those rumors have a tremendous effect on an individual’s reputation or societies. For example, more than 50% of the voter population had seen fake news in US 2016 election, and 50% of them believed in fake news stories [6]. Recently, Google has teamed up with Snopes.com [144] and Politifact.com [130] to validate and debunk rumor stories in OSNs. Current approaches (e.g., Snopes.com and Politifact.com) use human knowledge to manually label if a rumor is “False”, “Mostly False”, “True”, “Mostly True”, and “Half True”. As thousands of rumors are spread in OSNs in a very short time, manually labeling all those rumors is time-consuming and unrealistic. Recently, crowdsourcing solutions have been proposed to improve the results of machine learning tasks, such as machine translation [8] and sentiment analysis [152]. Some public crowdsourcing websites, such as Amazon Mechanical Turk, provide a mechanism to use human knowledge and insights to assign labels for some pre-defined tasks. However, each task has to be defined precisely, and the approach is not suitable for rumor labeling as rumors constantly change in real-time. To address this limitation, with a large number of users and their active participation, OSNs could effectively be a useful source of human input to debunk rumors [48]. Another limitation is that existing rumor veracity classification research only distinguishes if a rumor is “True”, “False”, and “Unverified”. We propose a newly-created rumor dataset with finer-grained truth levels (according to Snopes.com and Politifact.com) and use this dataset to study how early we could effectively identify the truth of rumors.

This chapter makes the following contributions:

- We introduce and analyze a dataset of 88 rumors from Reddit. Each rumor is identified as “False”, “Mostly False”, “True”, “Mostly True”, or “Half True”.
- We demonstrate that using established social science and psychology theories helps to select better feature sets for the rumor veracity detection task and provides a better understanding and detection of rumor veracity by integrating these theories with visualizations and machine learning techniques.
- We evaluate how early we could effectively identify rumor truth values and provide insights into breaking news and long-standing rumors. Our experimental results show that we could effectively detect the truth of rumors as early as seven days.

6.2 Related Work

6.2.1 Mining Rumor Data

One of the first publicly available rumor datasets is provided by Qazvinian et al. [132]. This dataset includes 10,000 tweets involving five different rumors. Each tweet is annotated as “related” or “unrelated” to a rumor. A dataset of 100 million tweets involving 72 rumors (41 true and 31 false) was constructed by Giasemidis et al. [67] and a machine learning approach was applied to it to classify whether those rumors are true or false. The PHEME dataset includes 1,972 rumorous and 3,830 non-rumorous tweets about five breaking news stories [51]. The dataset provided by Kwon et al. [96] is a collection of tweets for 61 rumors and 51 non-rumors, used to study how various feature sets affect the accuracy of rumor detection over time. The SemEval 2017 Task 8 [52] provides a dataset that includes tweets and an annotation label for each tweet, “support”, “deny”, “query”, or “comment”. Eight teams participated and submitted the results for this task. The winning system classified the stance of each tweet using features and labels of the previous tweets. As most of the existing datasets only focus on “false” and “true” rumors, we aim to provide a rumor dataset that could be used to identify the truthfulness of rumors in one of the five categories: “False”,

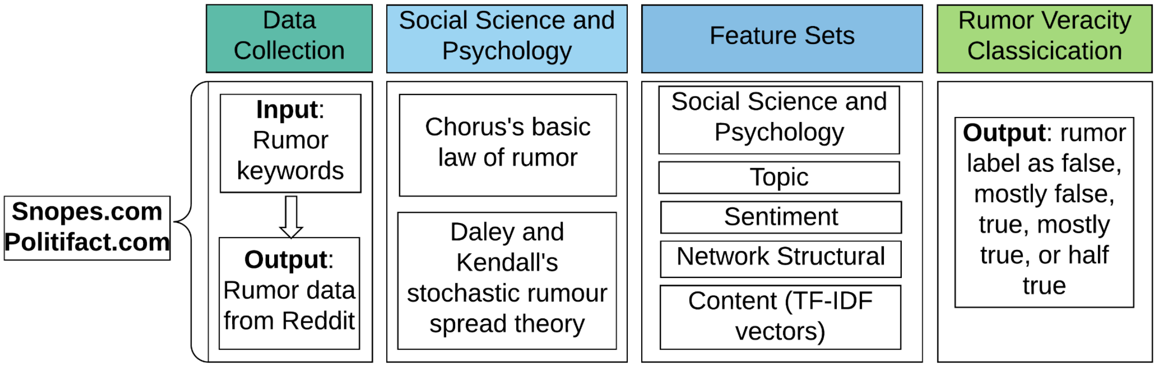

Data Collection
Social Science and Psychology
Feature Sets
Rumor Veracity Classicication
Snopes.com
Politifact.com
Input: Rumor keywords
Output: Rumor data from Reddit
Chorus's basic law of rumor
Daley and Kendall's stochastic rumour spread theory
Social Science and Psychology
Topic
Sentiment
Network Structural
Content (TF-IDF vectors)
Output: rumor label as false, mostly false, true, mostly true, or half true

Figure 6.1: Rumor veracity classification framework.

“Mostly False”, “True”, “Mostly True”, “Half True”. These fine-grained truth levels are used to reflect the nature of rumor spread in OSNs.

6.2.2 Rumor Analysis in OSNs

One of the first analyses about rumor spread was in 1944 by Festinger et al. [63]. The authors studied how rumors were spread in a particular neighborhood community. Due to the limitation of rumor data and the intrinsic long-lasting nature of rumors, rumor analysis was mostly theoretical research and experienced a long hiatus until the popularity of OSNs in the 2000s. In most OSNs, data is available, disseminated and stored permanently, so researchers have access to data to more efficiently study rumors and verify their theories. A classification approach to identify if a tweet is a rumor on Twitter was adopted by Qazvinian et al. [132]. Each tweet was manually assigned as either being related to rumors or not. Relations between claims associated with rumors and analyzed contradictory claims inside a rumor were interpreted by Lendvai and Reichel [99]. The credibility perceptions of rumors were studied by classifying if a tweet is related to a rumor into three classes: “certain”, “somewhat certain”, and “uncertain” in Zubiaga and Ji [169]. An autonomous message-classifier that filters relevant and trustworthy tweets was proposed in Giasemidis et al. [67]. How different feature sets could affect the performance of the rumor veracity task over time was studied in Kwon et al. [96]. Our work is different from those approaches as our starting point consists of established theories from social science and psychology. Using those concepts, we propose a new rumor dataset that better reflects various truth levels of a rumor. For the classification task, we use various feature sets that are derived based on the notion of how rumors are transmitted in OSNs.

6.3 Methodology — Rumor Veracity Classification Task

The definition of rumor veracity classification was first proposed by [168]. However, the authors only consider three-class labels: “False”, “True”, and “Unverified”. We extend that definition to our collected dataset with five class labels.

Let a rumor, RU_i, have a list of submissions in ascending time order SU_1, SU_2, ... SU_m and a list of topics T_1, T_2, ... T_n that are extracted from the submission titles in ascending time order. Each submission SU_j has a list of user comments C_1, C_2, ... C_k. Each user comment has a sentiment analysis score. If a comment C_x of user U_o replies to comment C_y of user U_p, there will be a connection from user U_o to user U_p in the user interaction graph. The task is to determine whether a rumor in OSNs could be categorized into one of the five categories: “False”, “Mostly False”, “True”, “Mostly True”, and “Half True” using the above submissions, user interaction graph, and other metadata.

6.3.1 Social Science and Psychology Features

Existing research has used various ad-hoc feature sets from rumor data for the rumor veracity classification task. The classification result is reported based on machine learning techniques without considering social science and psychology theories. Little work has tried to understand and build the feature sets from grounded theoretical work about rumor spread from social science and psychology. Linguistic Inquiry and Word Count to build a set of features that reflect the process of doubt, negation, and guessing of rumor propagation were used in [96, 55]. In contrast, we adopt two rumor theories. The first is the rumor spread theory “The Basic Law of Rumor” [35] where the truth and strength of a rumor depend on the importance of its topics and the significance of its ambiguity. For example, if a rumor is about an important individual (e.g., “Obama is a Muslim”), it is more likely to be spread further and more likely to be false. In addition, rumors that are hard to verify will likely last longer. Researchers have shown that the more controversial the comments, the more popular the post will be [49]. Intuitively, we take advantage of extracted Wikipedia topics from rumor data to represent the importance of a rumor (see Section 6.3.1) and the sentiment score of users’ comments to represent the controversy of a rumor (see Section 6.3.2) .

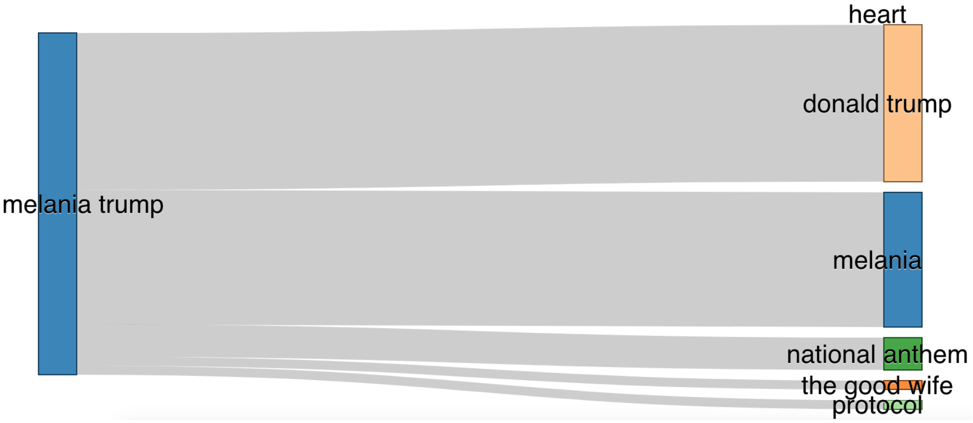
The second is the rumor spread model theory of Daley and Kendall [38]. We compute the numbers of *spreaders*, *ignorants*, and *stiflers* in a rumor throughout its life cycle as discussed in Dang et al. [45]. In this model, N is the number of users who interact with this rumor. In the beginning, one user learns about this rumor from another source and tries to spread it by posting a submission. Other users will read this submission and start spreading to other members. In each submission, a user is categorized into one of the three categories: spreaders, ignorants, and stiflers, which are denoted as $S, I,$ and R, respectively. Spreaders are people who actively spread the rumor; Ignorants are people who are ignorant of the rumor at first but will become either spreaders or stiflers at a later stage; Stiflers are people who posted about the rumor, but are no longer interested in spreading it. This rumor spread model is modeled as a stochastic process on $P = \{S, I, R\}^N$, where N is the total number of users for a rumor in the dataset. Let the state of user i at time t be a function of time $X_i(t)$. The procedure to compute $X_i(t)$ is as follows:

1. The user who posted the first submission about this rumor at time $t = 0$ is the spreader $(|S| = 1, |I| = N-1, |R| = 0)$.
2. Users who reply to the first submission are ignorants at t_0 and will become spreaders at t_1.
3. At time t_i:
   (a) If user j posts this submission, user j is a spreader.
   (b) If user j has a comment on this submission:
      i. User j is a spreader if user j has a comment at t_{i-1}.
      ii. User j is a stifler if user j has a comment at time from t_{i-2} to t_0.
      iii. User j is an ignorant if user j has no comments at time from t_{i-1} to t_0.
   (c) User j will become a spreader at time t_{i+1}.

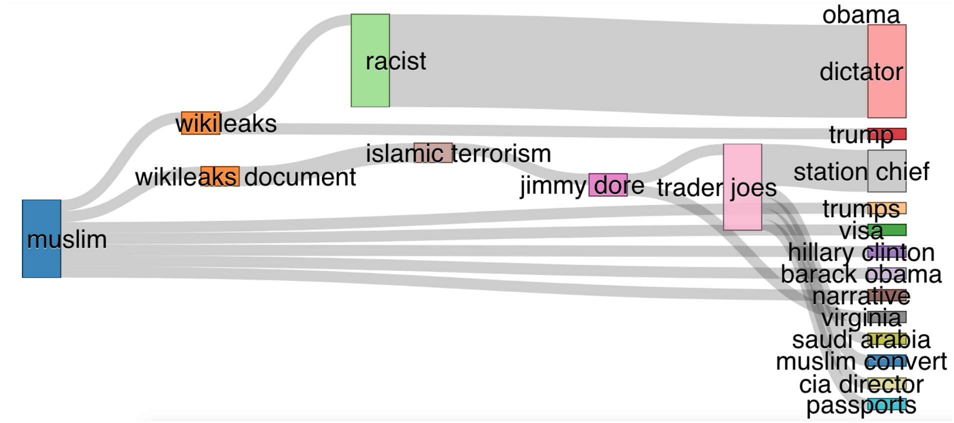
Based on those two rumor spread theories, we build various features for the rumor veracity classification task.

6.3.2 Topic Features

Previous studies highlight the importance of topics that affect the popularity of a rumor. The importance of a topic plays a significant role in the popularity of rumor spread according to Rosnow and Foster [138], while users spread rumors when they feel anxious about a topic they are interested in (e.g., AIDS-related rumors) according to Bordia and DiFonzo [20]. For each submission, we use the Dexter topic extraction tool, [149] to obtain all Wikipedia topics in submission titles. We use the number of topics in each rumor to determine how important this rumor is. Also, we compute the approximate entropy (ApEn) for each topic list of a rumor. ApEn is a method to evaluate the regularity and the unpredictability of the fluctuating nature of temporal data. Researchers have used this approach to compute topic



(a)



(b)

Figure 6.2: Topic features between true vs. false rumors: a) “Melania Trump Reminds the President to Put His Hand Over His Heart”, b) “Obama is a Muslim”. In this graph, each node is a Wikipedia topic that is extracted from the submission title. There is a connection between two continuous submissions of two topics if their semantic similarity score is above 0.5. For the true rumor, we have a concise list of topics, while we have a wide range of topics about the false rumor “Obama is a Muslim”. The node and connection size and color are generated automatically based on the number of topics for the best visual layout.

evolution models [18, 114]. Our first assumption is that true rumors are usually verified in a short time, while false rumors take longer to be debunked. The second assumption is that the topics of true rumors are more regular and predictable than false rumors. An example of topic evolution between a true rumor and a false rumor is shown in Figure 6.2. We observe that the topics of true rumors are more regular and less fluctuating than false rumors.

For each rumor RU_i, we have a list of topics $T_1, T_2, ...T_n$ in ascending order. To calculate the distance between two topics for ApEn, we compute the semantic similarity between two topics, T_1 and T_2, using Google Tri-gram Method [90].

6.3.3 Sentiment Features

Online rumors could draw attention, stimulate involvement, and influence attitudes and actions of OSN users [56]. In an online conversation, user sentiment significantly contributes to how news or topics become popular [49]. Each user comment is parsed into sentences and each sentence is assigned a sentiment score: “Positive”,“Negative”, or “Neutral” using the OpenNLP toolkit. We apply the concepts of sentiment polarity and subjectivity of Zhang and Skiena [163] for each rumor in our ground-truth dataset as follows:

$$polarity\_score = (p-n)/(p+n) \tag{6.1}$$

$$subjectivity\_score = (n+p)/N \tag{6.2}$$

where p is the number of positive statements, n is the number of negative statements, and N is the total number of statements (including neutral statements). Polarity score represents whether a rumor is associated with the entity positively or negatively, while subjectivity score depicts how much sentiment a rumor garners.

6.3.4 Network Structural Features

The questions of who spreads rumors and how have been studied extensively in the literature [57, 48]. Researchers have stated that rumors are usually spread by few influencing users, and these users could spread rumors a lot quicker and cause significant damage to the individual targets in OSNs [115]. Using this assumption, we compute the betweenness and closeness of the user interaction directed graph as shown in Figure 6.3 where a node is a user, and an edge between two nodes represents that a user (denoted by one node) replies to a comment of another user (denoted by another node). A user with a high betweenness centrality score could propagate rumorous news to a large user population in the network and influence the popularity of a rumor, while a user with a high closeness centrality score could transmit the rumor to a large user population in a short time. These users play a crucial role in detecting the truth of rumors in its early stage. As influencing users could significantly impact the popularity of a rumor and its truthfulness, we use the highest betweenness and closeness scores in the user graph as features for the classification task.

6.3.5 Content Analysis — the Wisdom of the Crowd

Previous studies have used various features to distinguish between rumors and non-rumors [132, 96]. In this chapter, we study whether we could identify the level of truth

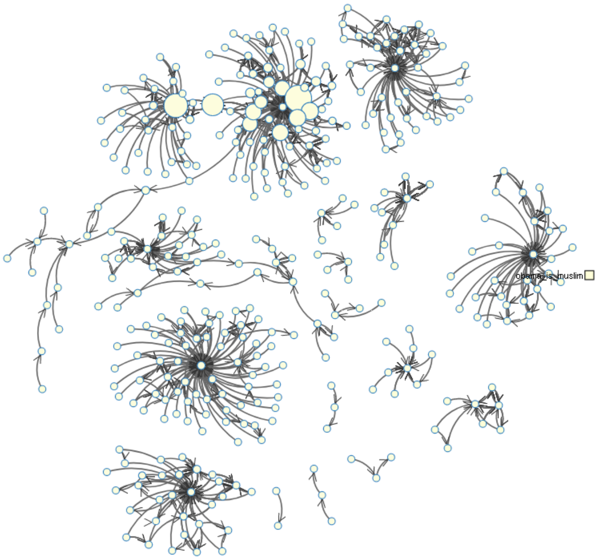


Figure 6.3: User interaction directed graph of the false rumor “Obama is a Muslim”. Each node is a user and an edge represents the connection between two users (who replies to who). The node size represents the centrality score of the users.

of rumors based on how users respond to these rumors. Why people spread rumors in a social network psychologically has been studied [20]. People spread rumors based on three motivations: (1) fact-finding, (2) relationship-building, and (3) self-enhancement. Fact-finding people intend to find the truth of rumors through a problem-solving process. In contrast, those motivated by relationship-building are simply interested in communicating and interacting with other individuals by sharing information about particular rumors. Self-enhancement people are either consciously or unconsciously approving false rumors. How users interact with each other within a rumor was studied in Dang et al. [48], finding that a dominant number of users just try to joke about this rumor. Also, there are more users who try to disapprove a false rumor than to approve it. Based on this finding, we aim to identify whether we could use the wisdom of the crowd (i.e., users’ comments) to debunk a rumor and find its truth. Social science and Psychology, Topic, Sentiment, Network Structural, and Content features for the rumor veracity classification task are shown in Table 6.1. The architecture of the proposed approach is shown in Figure 6.1.

6.4 Results

This section describes the analysis of the data and highlights the characteristics of rumors in each category using the experimental results. We first report the results on how to accurately classify rumors into one of the five categories. Next, we treat the rumors in the

Table 6.1: Feature sets that are established from ground-up social science theories and other Basic Features (BF).

Group	Features
Social Science and Psychology	Number of spreaders Number of stiflers Number of ignorants
Topic	Number of topics Approximate entropy score
Sentiment	Polarity score Subjectivity score
Network Structural	Number of submissions (BF) Number of comments (BF) Number of unique users (BF) Betweenness centrality score Closeness centrality score
Content	TF-IDF user comment vectors

"Mostly False" to be the same as the category "False", and the rumors in the "Mostly True" to be the same as the category "True". This combination results in a three-class classification task: "False", "True", and "Half True". Finally, we filter out the rumors in the category "Half True" and report the result for a two-class classification task.

6.4.1 The Newly-created Reddit Rumor Dataset

For each rumor, we need to collect the following elements: 1) The truth about this rumor, 2) Posts (data) about this rumor, and 3) Metadata about this rumor, such as sentiment analysis, topics, and user interaction graphs. For the dataset, we first identified the most popular rumors (58 true and false rumors) from previous work [96, 52]. In addition, we collect 30 new rumors from Snopes.com and Politifact.com that could belong to one of the three new categories: "Mostly False", "Mostly True", and "Half True". The labels of the combined rumor list are verified with the five categories from these websites. We identified

Table 6.2: The newly-created Reddit rumor ground-truth dataset. The table also shows the number of long-standing rumors in each category.

Category	No. Rumors	Avg. No. Submissions	Avg. No. Comments	Long-Standing Rumors
False	48	14	249	34
Mostly False	10	11	198	7
True	10	8	98	7
Mostly True	10	87	8	8
Half True	10	6	99	8

Table 6.3: Classification results for different sets of categories.

Category	Accuracy	Precision	Recall	F_1
Five classes	0.589	0.584	0.545	0.564
Three classes	0.670	0.670	0.670	0.670
Two classes	0.752	0.750	0.744	0.747

a total of 88 rumors (see the Supplementary Material — Rumor Description) and extracted submissions if a submission contains explicit keywords relevant to the rumor. We adopt RumourFlow [45] to collect and visualize rumor data in Reddit. All collected rumor data are provided to the visualization system through a JSON restful web service and a JAVA backend. The goal of using RumourFlow visualization is to analyze and understand the characteristics of rumors in each category and confirm whether the feature sets derived from social science work could be applied.

Overall, we collected 88 rumors that belong to one of the five categories: “False”, “Mostly False”, “True”, “Mostly True”, and “Half True”. For each rumor category, we also report the number of long-standing rumors vs. breaking news rumors. Long-standing rumors are rumors that have been discussed and propagated for a long period while breaking news rumors are usually circulated in breaking news events, such as natural disaster, political events in their early circulation [168]. The long-standing rumors are dominant in the

Table 6.4: Classification results for topic, sentiment and structural features.

Category	Accuracy	Precision	Recall	F_1
Five classes	0.484	0.442	0.398	0.419
Three classes	0.631	0.546	0.534	0.540
Two classes	0.781	0.674	0.628	0.650

Table 6.5: Classification results for combined features of content, topic, sentiment and structural features.

Category	Accuracy	Precision	Recall	F_1
Five classes (5C)	0.795	0.586	0.729	0.761
5C minus structural	0.593	0.593	0.729	0.654
Three classes (3C)	0.864	0.860	0.852	0.856
3C minus minus structural	0.745	0.745	0.792	0.768
Two classes (2C)	0.927	0.928	0.879	0.903
2C minus structural	0.795	0.796	0.729	0.761

dataset. A detailed statistics of the newly-created rumor dataset is shown in Table 6.2. We observed that false rumors and mostly false rumors receive the highest number of discussed submissions and comments. This supports the assumption that false rumors are more popular than true rumors [115].

6.4.2 Content Feature Classification

We transform users' comments on rumors into TF-IDF vectors, the elements of which reflect how important a word is in a document or in a corpus (stop words are removed). Each rumor is represented by a vector of the 200 highest ranking words. We choose Naive Bayes classification (NB) with ten-fold cross validation from Weka [76], as NB is fast to build and could be trained with less data. Those two characteristics of NB are important for the rumor veracity task. The results of the classification are shown in Table 6.3.

For the five class result, we achieve 56.4% F_1 score. The results get better with three classes and two classes with 67% and 74.7% F_1 score respectively. The result shows that the proposed approach can better distinguish the difference between true rumors and false rumors. We achieve the best result without the category "Half True". This is because "Half True" rumors are very controversial and not easily identifiable.

6.4.3 Social Science and Psychology, Topic, Sentiment, and Network Structural Feature Classification Result

We report the classification results using social science and psychology, topic, sentiment, and network structural features in Table 6.4. We have the same pattern as using the content features, but the results do not perform as well as using the content features. The classification results of two and three classes are better than the five classes in terms of accuracy and recall. We again observe that the classification results are significantly better for the two-class classification task. This supports that the rumors in the category "Half True" are harder to detect. Although the system achieved lower accuracy than the content features, the results are still close. We plan to integrate the intrinsic characteristics of the two feature sets aiming to achieve a better result than either.

6.4.4 Combined Feature Classification Result

After combining the social science and psychology, sentiment, topic, and network structural features with the 200 highest-ranking attributes of content features from TF-IDF vectors, we achieve the best result as shown in Table 6.5. We achieve the best precision and recall using only two classes, and this result is comparable with the two-class veracity classification results reported in Kwon et al. [96] on a different dataset. We also evaluate the importance of the three social science features: number of spreaders, number of stiflers, and number of ignorants from the social science and psychology group by performing the ablation test. The three attributes that are built from Daley and Kendall's stochastic rumor spread theory are demonstrated to significantly improve classification quality, as shown in Table 6.5.

6.4.5 Rumor Truth Time-varying Result

We also investigate whether we can detect a rumor in its earlier stages and still maintain accuracy. As rumors may have different peak cycles, we build the combination feature sets based on different time windows and classify using the following intervals: three hours, three days, seven days, 14 days, and 28 days. Classification achieves an F_1 score above 60% after the first seven days and the result after 28 days is comparable to the best result in Table 6.5, as shown in Figure 6.4. The system did not perform well when trying to find the level

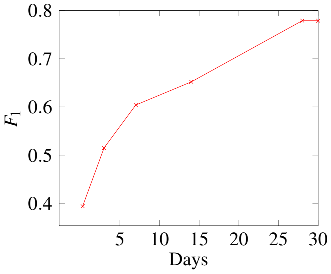
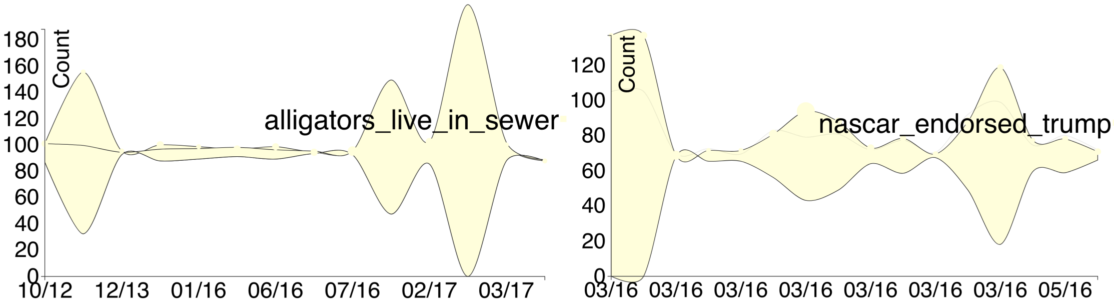


Figure 6.4: Five-class classification results over time. The system achieves the best F_1 score (0.78) on day 28.



(a)

(b)

Figure 6.5: Long-standing vs. breaking news rumors a) The long-standing false rumor “(alligators live in sewers) ” has various peaks and the highest peak is usually not the first peak, b) The breaking news rumor “Nascar endorsed Trump” has the highest peak first and smaller peaks occurred later. The x-axis represents the time, while the y-axis represents the number of comments for each submission.

of truth of rumors after three hours or three days.

We observe and compare the fluctuating nature between long-standing vs. breaking news rumors using RumourFlow in Figure 6.5. The long-standing rumors have various peaks over a long period, and the highest peak of comments is usually reached after the rumor has been discussed for a while. On the other hand, the breaking news rumors usually have the highest peak at the beginning and several smaller peaks until they die. We select rumor data from the beginning until the highest peak of comments. For the first peak, we select rumor data from the beginning until the first peak of comments. Using this finding, we investigate the performance of the veracity classification task after a rumor's first peak and highest peak. For the five-class veracity classification task, we achieve the best F_1 = 0.78 using rumors' highest peak and F_1= 0.73 using the rumors' first peak. This is an important finding as the system could effectively detect the level of truth for breaking news rumors in a short time (the average highest peak of breaking news rumors in the dataset is three hours).

6.5 Discussion and Conclusion

In this chapter, we introduce a new Reddit rumor dataset where each rumor is categorized into one of five categories: “False”, “Mostly False”, “True”, “Mostly True”, and “Half True”. The truth levels of rumors in this dataset better reflect the fine-tuned truth values of rumors in Snopes.com and Politifact.com. Next, we investigate whether the proposed approach can effectively detect and debunk the truthfulness of rumors through an extensive set of features including social science and psychology theories. The experiments show that our system can efficiently detect the truthfulness of rumors. This could bridge the gap between social science theories and experimental research of rumor spread in online social networks.

We also explore various feature sets and levels of truth of rumors in the experiments. We found that the system best detects rumors in two classes “False” or “True”. The “Half True” category degrades the classification result. One of the underlying reasons is the conflicting characteristics of such rumors. We also found similarity between “False” and “Mostly False” as well as between “True” and “Mostly True” rumor categories. This shows that social media users do not distinguish between those two categories. This observation matches the characteristics of political rumors where politicians do not always make 100% true statements. Our two-class rumor veracity classification result is comparable with the the-state-of-the-art method in the literature [96]. The experiments also show that the attributes of social science theories significantly boost the result of the rumor veracity task.

Early detection of the truth of rumors is a key factor in preventing their spread. The experiments show that we could effectively debunk rumors as early as in seven days. We also find that the proposed approach can efficiently find the truth of rumors after their first peak ($F_1 = 0.73$). Hence, it is possible to effectively detect the truth levels of breaking news rumors in three hours. On the other hand, the long-standing rumors could be efficiently debunked after the rumors’ highest peaks (usually after 28 days). Due to Twitter API limits and lack of availability of sufficient data in the existing datasets in the literature, we have not been able to construct and evaluate our proposed feature set derived from social sciences and psychology with those datasets. Important future work will be to extend and compare the results of the proposed Reddit rumor dataset with other comparable rumor datasets (e.g., rumors on Twitter). In addition, solving the problem of the “Half True” rumors is an urgent need as it is more and more popular in political news.

Chapter 7

Conclusions

In this thesis, we first introduce a new Reddit temporal n-gram corpus, which is designed specifically for social media text. We create the corpus using distributed parallel computing and implement a cloud-based visualization interface so that end users can query any n-grams from the corpus. Both the corpus and the interface are publicly available in this URL - Reddit n-gram temporal corpus. This large-scale terabyte corpus includes all the word unigram to 5-gram and their frequency per month from October 2007 to August 2016. Using the 5-gram in the corpus, we introduce a topic based latent semantic analysis to compute semantic similarity for social media texts. The proposed topic-based latent semantic analysis outperforms all the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015 Task 1 — Semantic Similarity for the PIT-2015 dataset. Combined with sentiment analysis, the proposed approach also achieves the best result for the Paraphrase Identification of SEMEVAL 2015 Task 1. In addition, TLSA is language-independent and scalable for the large-scale nature of social media text.

First, we present a meme detecting framework to identify emerging events and rumour-driven topics in online social networks. This framework makes use of semantic similarity and Wikipedia concepts as external knowledge for the meme clustering task. It also defines several pairwise similarity scores between elements of two submissions. These strategies include average, maximum, linear combination, internal centrality-based weighting, and similarity score reweighting with relevance user feedback. The internal centrality-based weighting strategy computes the weight factors for each element of a submission by considering its surrounding context, while the similarity score reweighting with relevance user feedback strategy involves end-users to give a score between two submissions. Finally, we propose an offline-online meme clustering framework to both detect memes in real time and achieve good clustering results.

Second, we proposed RumorFlow, a visualization framework for visualizing rumour evolution, topic change, and user activity in online social networks. End users can collect, analyze, and visualize various aspects that are typical of rumour spreading, such as simultaneous analysis of user and topic, rumour and topic, and rumour and user. Social science research and analysis could benefit from the system and the approach. It is our intention to create a layer of an application-specific interface to allow access by typical users in such human behavioral studies.

Finally, using the proposed visualization framework, we analyze rumour spread patterns and introduce a novel approach to detect rumour veracity in its early stage. Each rumour is categorized into one of five categories: “False”, “Mostly False”, “True”, “Mostly True”, and “Half True”. The truth levels of rumours in this dataset better reflect the fine-tuned truth values of rumours in Snopes.com and Politifact.com. The system adopts an extensive set of features including social science and psychology theories for the rumour veracity detection task. The experiments show that our system can efficiently detect the truthfulness of rumours.

The end goals of the thesis are to study and improve two limitations of online social networks. For the first limitation, we try to improve the shortness and noisiness of social media text that is transmitted in online social networks. The results in Chapter 2 show that the newly-created and large-scale n-gram corpus that is derived from social media data could improve the performance of current natural language processing systems. This contribution provides opportunities for existing natural language processing systems to overcome the intrinsic limitation of social media text and advance the-state-of-the-art NLP systems. For the second limitation, we aim to limit the spread of misleading information that is spread in online social networks by collecting, analyzing, visualizing, detecting, and debunking rumors in online social networks. The results from Chapter 3, Chapter 4, Chapter 5, and Chapter 6 show that the proposed approach could effectively detect and debunk rumors in its early stage. With this contribution, the thesis could play an important role in promoting the spread of credible information and limiting the spread of false information in online social networks.

Another crucial contribution of the research is to provide a different angle for rumour research based on social science theories. The results from Chapter 4, Chapter 5, and Chapter 6 show that using knowledge and insights from established social science theories provides a better understanding of rumour spread and patterns in online social networks and improves the classification results for rumour detection. This could bridge the gap between social science theories and experimental research of rumour spread in online social networks.

7.1 Future Research Directions

For future work, we aim to take advantage of the newly-create corpus to study the linguistic patterns of social media text and finding the meaning of new words in social media. We also plan to integrate this proposed semantic similarity algorithm into our existing work for the meme clustering tasks and proposed visualization framework.

Another crucial future work is to extend the proposed visualization framework to other social network websites, such as Twitter, Facebook, and Google Plus. This will help researchers to understand the patterns of how a rumour is spread, its pattern, and detect emerging rumours. Comparing the spread of rumour-driven memes between Reddit and other OSNs and finding a correlation between them will provide a more holistic view of rumour spread.

Important future work will be to extend and compare the results of the proposed Reddit rumor dataset with other rumor datasets (e.g., rumors on Twitter). In addition, solving the problem of the “Half True” rumors is an urgent need as it is more and more popular in political news.

Investigative applications of various kinds can also benefit from the proposed framework. For general end-users, we intend to build a set of training-by-example tools and video tutorials for human learning. All related datasets and documentation will be publicly available. Two particular important extensions are planned. One involves using the system to inspire and extract features for categorization of user behaviors towards rumors, and a second extension involves adding the possibility for users to remove and add information, influencing the layouts. Topics, for instance, are currently extracted with Wikipedia concepts, but users will be able to suggest topics, that can then be detected and added to the current visualizations. The system handles large amounts of data at once in near real time, except for the preprocessing phase. However, some networks reach tens of thousands of users and hundreds of thousands of comments, users and for those, we intend to build a parallel processing framework so as to analyze rumor in massive user networks.

Bibliography

[1] Charu C Aggarwal, Jiawei Han, Jianyong Wang, and Philip S Yu. A framework for clustering evolving data streams. In *Proceedings of the 29th International Conference on very large Databases-Volume 29*, pages 81–92. VLDB Endowment, 2003.

[2] Charu C Aggarwal and Karthik Subbian. Event detection in social streams. In *Proceedings of the 2012 SIAM International Conference on Data Mining*, pages 624–635. SIAM, 2012.

[3] Charu C Aggarwal, Yuchen Zhao, and Philip S Yu. On clustering graph streams. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 478–489. SIAM, 2010.

[4] Rakesh Agrawal, Sreenivas Gollapudi, Krishnaram Kenthapadi, Nitish Srivastava, and Raja Velu. Enriching textbooks through data mining. In *Proceedings of the First ACM Symposium on Computing for Development*, page 19. ACM, 2010.

[5] Conrad Albrecht-Buehler, Benjamin Watson, David Shamma, et al. Textpool: Visualizing live text streams. In *IEEE Symposium on Information Visualization.*, pages p1–p1, 2004.

[6] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. Technical report, National Bureau of Economic Research, 2017.

[7] Jamal Alsakran, Yang Chen, Ye Zhao, Jing Yang, and Dongning Luo. Streamit: Dynamic visualization and interactive exploration of text streams. In *IEEE Pacific Visualization Symposium.*, pages 131–138, 2011.

[8] Vamshi Ambati, Stephan Vogel, and Jaime G. Carbonell. Active learning and crowdsourcing for machine translation. In *Language Resources and Evaluation*, 2010.

[9] Somnath Banerjee, Krishnan Ramanathan, and Ajay Gupta. Clustering short texts using Wikipedia. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 787–788. ACM, 2007.

[10] Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226, 2009.

[11] Alberto Barrón-Cedeño, Marta Vila, M Antònia Martí, and Paolo Rosso. Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4):917–947, 2013.

[12] Jason Baumgartner. Complete public Reddit comments corpus, July, 2015. Available: https://archive.org/details/2015_reddit_comments_corpus [Accessed: April 13, 2016].

[13] Hila Becker, Mor Naaman, and Luis Gravano. Learning similarity metrics for event identification in social media. In *Proceedings of the Third ACM International Conference on Web search and Data Mining*, pages 291–300. ACM, 2010.

[14] Kelly Bergstrom. Don't feed the troll: Shutting down debate about community expectations on Reddit.com. *First Monday*, 16(8), 2011.

[15] Pavel Berkhin. A survey of clustering data mining techniques. In *Grouping Multidimensional Data*, pages 25–71. Springer, 2006.

[16] Mikhail Bilenko, Sugato Basu, and Raymond J Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning*, page 11. ACM, 2004.

[17] William Blacoe and Mirella Lapata. A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on EMNLP and Computational Natural Language Learning*, pages 546–556. ACL, 2012.

[18] David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 113–120. ACM, 2006.

[19] Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka. Measuring semantic similarity between words using web search engines. *WWW*, 7:757–766, 2007.

[20] Prashant Bordia and Nicholas DiFonzo. Psychological motivations in rumor spread. *Rumor Mills: The Social Impact of Rumor and Legend*, pages 87–101, 2005.

[21] Thorsten Brants and Alex Franz. The Google Web 1T 5-gram corpus version 1.1. *Technical Report*, 2006.

[22] Thorsten Brants and Alex Franz. {Web 1T 5-gram Version 1}. 2006.

[23] Thorsten Brants and Alex Franz. Web 1T 5-gram, 10 european languages version 1. *Linguistic Data Consortium, Philadelphia*, 2009.

[24] Ceren Budak, Divyakant Agrawal, and Amr El Abbadi. Limiting the spread of misinformation in social networks. In *Proceedings of the 20th International Conference on World Wide Web*, pages 665–674. ACM, 2011.

[25] Cody Buntain and Jennifer Golbeck. Identifying social roles in Reddit using network structure. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, pages 615–620. International World Wide Web Conferences Steering Committee, 2014.

[26] Mario Cataldi, Luigi Di Caro, and Claudio Schifanella. Emerging topic detection on Twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, page 4. ACM, 2010.

[27] Bruce D Caulkins, Joohan Lee, and Morgan Wang. A dynamic data mining technique for intrusion detection systems. In *Proceedings of the 43rd Annual Southeast Regional Conference-Volume 2*, pages 148–153. ACM, 2005.

[28] Diego Ceccarelli, Claudio Lucchese, Salvatore Orlando, Raffaele Perego, and Salvatore Trani. Dexter 2.0-an open source tool for semantically enriching data. In *International Semantic Web Conference (Posters & Demos)*, pages 417–420, 2014.

[29] Ching-Yun Chang and Stephen Clark. Linguistic steganography using automatically generated paraphrases. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 591–599. Association for Computational Linguistics, 2010.

[30] Joong Hyuk Chang and Won Suk Lee. Finding recent frequent itemsets adaptively over online data streams. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 487–492. ACM, 2003.

[31] Cynthia Chew and Gunther Eysenbach. Pandemics in the age of Twitter: content analysis of tweets during the 2009 h1n1 outbreak. *PloS one*, 5(11):e14118, 2010.

[32] Yun Chi, Haixun Wang, Philip S Yu, and Richard R Muntz. Moment: Maintaining closed frequent itemsets over a stream sliding window. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pages 59–66. IEEE, 2004.

[33] Flavio Chierichetti, Ravi Kumar, Sandeep Pandey, and Sergei Vassilvitskii. Finding the Jaccard median. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 293–311. Society for Industrial and Applied Mathematics, 2010.

[34] Jaegul Choo, Hanseung Lee, Zhicheng Liu, John Stasko, and Haesun Park. An interactive visual testbed system for dimension reduction and clustering of large-scale high-dimensional data. In *IS&T/SPIE Electronic Imaging*, pages 865402–865402. International Society for Optics and Photonics, 2013.

[35] A Chorus. The basic law of rumor. *The Journal of Abnormal and Social Psychology*, 48(2):313, 1953.

[36] Jason Chuang, Christopher D Manning, and Jeffrey Heer. Termite: Visualization techniques for assessing textual topic models. In *International Working Conference on Advanced Visual Interfaces*, pages 74–77. ACM, 2012.

[37] Charles LA Clarke, Gordon V Cormack, M Laszlo, Thomas R Lynam, and Egidio L Terra. The impact of corpus size on question answering performance. In *ACM SIGIR*, pages 369–370. ACM, 2002.

[38] D. J. Daley and D. G. Kendall. Stochastic Rumors. *Ima Journal of Applied Mathematics*, 1:42–55, 1965.

[39] A. Dan and J. Richards. Behind the rumors: how we build our Twitter riots interactive @ONLINE, 2011. Available: http://www.guardian.co.uk/news/datablog/2011/dec/08/twitter-riots-interactive [Accessed: April 15, 2015].

[40] Anh Dang, Raheleh Makki, Abidalrahman Moh'd, Aminul Islam, Vlado Keselj, and Evangelos E Milios. Real time filtering of tweets using Wikipedia concepts and Google tri-gram semantic relatedness. In *TREC*, 2015.

[41] Anh Dang, Abidalrahman Moh'd, Anatoliy Gruzd, Evangelos Milios, and Rosane Minghim. A visual framework for clustering memes in social media. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pages 713–720. ACM, 2015.

[42] Anh Dang, Abidalrahman Moh'd, Anatoliy Gruzd, Evangelos Milios, and Rosane Minghim. A visual framework for clustering memes in social media. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, pages 713–720, New York, NY, USA, 2015. ACM.

[43] Anh Dang, Abidalrahman Moh'd, Anatoliy Gruzd, Evangelos Milios, and Rosane Minghim. An offline–online visual framework for clustering memes in social media. In *From Social Data Mining and Analysis to Prediction and Community Detection*, pages 1–29. Springer International Publishing, 2017.

[44] Anh Dang, Abidalrahman Moh'd, Evangelos Milios, and Rosane Minghim. What is in a rumor: Combined visual analysis of rumor flow and user activity. In *Proceedings of the 33rd Computer Graphics International*, pages 17–20. ACM, 2016.

[45] Anh Dang, Abidalrahman Moh'd, Evangelos Milios, and Rosane Minghim. What is in a rumour: Combined visual analysis of rumour flow and user activity. In *Proceedings of the 33rd Computer Graphics International*, pages 17–20. ACM, 2016.

[46] Anh Dang, Abidalrahman Moh'd, Aminul Islam, and Evangelos Milios. Early detection of rumor veracity in social media. In *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 2019.

[47] Anh Dang, Abidalrahman Moh'd, Aminul Islam, Rosane Minghim, Michael Smit, and Evangelos Milios. Reddit temporal n-gram corpus and its applications on paraphrase and semantic similarity in social media using a topic-based latent semantic analysis. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3553–3564, 2016.

[48] Anh Dang, Michael Smit, Abidalrahman Moh'd, Rosane Minghim, and Evangelos Milios. Toward understanding how users respond to rumours in social media. In *Advances in Social Networks Analysis and Mining (ASONAM), 2016 IEEE/ACM International Conference on*, pages 777–784. IEEE, 2016.

[49] Linh Dang-Xuan and Stefan Stieglitz. Impact and diffusion of sentiment in political communication-an empirical analysis of political weblogs. In *International Conference on Web and Social Media*, 2012.

[50] Dipanjan Das and Noah A Smith. Paraphrase identification as probabilistic quasi-synchronous recognition. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1*, pages 468–476. Association for Computational Linguistics, 2009.

[51] Leon Derczynski and Kalina Bontcheva. Pheme: Veracity in digital social networks. In *UMAP Workshops*, 2014.

[52] Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. *arXiv preprint arXiv:1704.05972*, 2017.

[53] B Dervin. An overview of sense-making research: Concepts. *Methods, and Results to Date [on-line] Disponível and Internet*, 1983.

[54] Michel Marie Deza and Elena Deza. *Encyclopedia of Distances*. Springer, 2009.

[55] Nicholas DiFonzo and Prashant Bordia. Corporate rumor activity, belief and accuracy. *Public Relations Review*, 28(1):1–19, 2002.

[56] Nicholas DiFonzo and Prashant Bordia. *Rumor psychology: Social and organizational approaches.* American Psychological Association, 2007.

[57] Benjamin Doerr, Mahmoud Fouz, and Tobias Friedrich. Why rumors spread so quickly in social networks. *Communications of the ACM*, 55(6):70–75, 2012.

[58] Asli Eyecioglu and Bill Keller. Asobek: Twitter paraphrase identification with simple overlap features and SVMs. *SemEval*, 2015.

[59] Tanja Falkowski, Jorg Bartelheimer, and Myra Spiliopoulou. Mining and visualizing the evolution of subgroups in social networks. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 52–58. IEEE Computer Society, 2006.

[60] FEMA. Hurricane Sandy: Rumor control @ONLINE, 2012. Available: http://neteffect.foreignpolicy.com/posts/2009/04/25/swine_flu_twitters _power_to_misinform [Accessed: April 15, 2015].

[61] Samuel Fernando and Mark Stevenson. A semantic similarity approach to paraphrase detection. In *Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, pages 45–52. Citeseer, 2008.

[62] Adriano Ferraresi, Silvia Bernardini, Giovanni Picci, and Marco Baroni. Web corpora for bilingual lexicography: a pilot study of english/french collocation extraction and translation. *Using Corpora in Contrastive and Translation Studies. Newcastle: Cambridge Scholars Publishing*, pages 337–362, 2010.

[63] Leon Festinger, Dorwin Cartwright, Kathleen Barber, Juliet Fleischl, Josephine Gottsdanker, Annette Keysen, and Gloria Leavitt. A study of a rumor: its origin and spread. *Human Relations*, 1948.

[64] Adrien Friggeri, Lada Adamic, Dean Eckles, and Justin Cheng. Rumor cascades. In *Eighth International AAAI Conference on Weblogs and Social Media*, 2014.

[65] Emden R Gansner, Yifan Hu, and Stephen North. Visualizing streaming text data with dynamic graphs and maps. In *Graph Drawing*, pages 439–450. Springer, 2013.

[66] Chris Giannella, Jiawei Han, Jian Pei, Xifeng Yan, and Philip S Yu. Mining frequent patterns in data streams at multiple time granularities. *Next Generation Data Mining*, 212:191–212, 2003.

[67] Georgios Giasemidis, Colin Singleton, Ioannis Agrafiotis, Jason RC Nurse, Alan Pilgrim, Chris Willis, and Danica Vukadinovic Greetham. Determining the veracity of rumours on Twitter. In *International Conference on Social Informatics*, pages 185–205. Springer, 2016.

[68] Karam Gouda and Mohammed Javeed Zaki. Efficiently mining maximal frequent itemsets. In *Proceedings 2001 IEEE International Conference on Data Mining*, pages 163–170. IEEE, 2001.

[69] Mark Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6):1420–1443, 1978.

[70] Mark S Granovetter. The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1380, 1973.

[71] Patricia M Greenfield. The changing psychology of culture from 1800 through 2000. *Psychological Science*, 24(9):1722–1731, 2013.

[72] Brynjar Gretarsson, John O'donovan, Svetlin Bostandjiev, Tobias Höllerer, Arthur Asuncion, David Newman, and Padhraic Smyth. Topicnets: Visual analysis of large text corpora with topic modeling. *ACM Transactions on Intelligent Systems and Technology*, 3(2):23, 2012.

[73] Weiwei Guo and Mona Diab. Modeling sentences in the latent space. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers- Volume 1*, pages 864–872. ACL, 2012.

[74] Weiwei Guo, Wei Liu, and Mona Diab. Fast tweet retrieval with compact binary codes. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 486–496, 2014.

[75] Maria Halkidi. Quality assessment and uncertainty handling in data mining process. In *EDBT PhD Workshop*, 2000.

[76] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[77] Lushan Han, Abhay L Kashyap, Tim Finin, James Mayfield, and Jonathan Weese. Umbc_ebiquity-core: Semantic textual similarity systems. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, volume 1, pages 44–52, 2013.

[78] Samer Hassan Hassan and Rada Mihalcea. Semantic relatedness using salient semantic analysis. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[79] Susan Havre, Elizabeth Hetzler, Paul Whitney, and Lucy Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics.*, 8(1):9–20, 2002.

[80] Amaç Herdağdelen and Marco Baroni. Stereotypical gender actions can be extracted from web text. *Journal of the American Society for Information Science and Technology*, 62(9):1741–1749, 2011.

[81] Edward Herman and Noam Chomsky. *Manufacturing consent: The political economy of the mass media*. Random House, 2010.

[82] Francis Heylighen. What makes a meme successful? selection criteria for cultural evolution. 1998.

[83] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in Artificial Intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc., 1999.

[84] Danny Holten and Jarke J Van Wijk. Force-directed edge bundling for graph visualization. In *Computer Graphics Forum*, volume 28, pages 983–990. Wiley Online Library, 2009.

[85] Liangjie Hong and Brian D Davison. Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics*, pages 80–88. ACM, 2010.

[86] E. Hoque and G. Carenini. Convis: A visual text analytic system for exploring blog conversations. *Computer Graphics Forum*, 33(3):221–230, June 2014.

[87] Mengdie Hu, Shixia Liu, Furu Wei, Yingcai Wu, John Stasko, and Kwan-Liu Ma. Breaking news on Twitter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2751–2754. ACM, 2012.

[88] Xiaohua Hu, Xiaodan Zhang, Caimei Lu, E. K. Park, and Xiaohua Zhou. Exploiting Wikipedia as external knowledge for document clustering. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 389–396, New York, NY, USA, 2009. ACM.

[89] Aminul Islam, Jie Mei, Evangelos E Milios, and Vlado Kešelj. When was Macbeth written? mapping book to time. In *Computational Linguistics and Intelligent Text Processing*, pages 73–84. Springer, 2015.

[90] Aminul Islam, Evangelos Milios, and Vlado Kešelj. Text similarity using Google tri-grams. In *Proceedings of the 25th Canadian Conference on Advances in Artificial Intelligence*, Canadian AI'12, pages 312–317, Berlin, Heidelberg, 2012. Springer-Verlag.

[91] Mohsen JafariAsbagh, Emilio Ferrara, Onur Varol, Filippo Menczer, and Alessandro Flammini. Clustering memes in social media streams. *Social Network Analysis and Mining*, 4(1), 2014.

[92] George Jennifer. *Understanding and managing organizational behavior*. Pearson Education India, 2009.

[93] Yangfeng Ji and Jacob Eisenstein. Discriminative improvements to distributional sentence similarity. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 891–896, 2013.

[94] Halil Kiymaz. The stock market rumours and stock prices: a test of price pressure and size effect in an emerging market. *Applied Financial Economics*, 12(7):469–474, 2002.

[95] 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*, pages 591–600. AcM, 2010.

[96] Sejeong Kwon, Meeyoung Cha, and Kyomin Jung. Rumor detection over varying time windows. *PLOS ONE*, 12(1):e0168344, 2017.

[97] Thomas K Landauer, Peter W Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3):259–284, 1998.

[98] Hanseung Lee, Jaeyeon Kihm, Jaegul Choo, John Stasko, and Haesun Park. ivisclustering: An interactive visual document clustering via topic modeling. In *Computer Graphics Forum*, volume 31, pages 1155–1164. Wiley Online Library, 2012.

[99] Piroska Lendvai and Uwe D Reichel. Contradiction detection for rumorous claims. *arXiv preprint arXiv:1611.02588*, 2016.

[100] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 497–506. ACM, 2009.

[101] Stephan Lewandowsky, Ullrich KH Ecker, Colleen M Seifert, Norbert Schwarz, and John Cook. Misinformation and its correction continued influence and successful debiasing. *Psychological Science in the Public Interest*, 13(3):106–131, 2012.

[102] Chen Li and Yang Liu. Improving named entity recognition in tweets via detecting non-standard words. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 929–938, 2015.

[103] Hua-Fu Li, Suh-Yin Lee, and Man-Kwan Shan. An efficient algorithm for mining frequent itemsets over the entire history of data streams. In *Proceedings of First International Workshop on Knowledge Discovery in Data Streams*, volume 39, 2004.

[104] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *journal of the Association for Information Science and Technology*, 58(7):1019–1031, 2007.

[105] Fei Liu, Fuliang Weng, and Xiao Jiang. A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 1035–1044. Association for Computational Linguistics, 2012.

[106] Linyuan Lü and Tao Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, 2011.

[107] Michal Lukasik, Trevor Cohn, and Kalina Bontcheva. Classifying tweet level judgements of rumors in social media. *ArXiv Preprint ArXiv:1506.00468*, 2015.

[108] Michal Lukasik, PK Srijith, Duy Vu, Kalina Bontcheva, Arkaitz Zubiaga, and Trevor Cohn. Hawkes processes for continuous time sequence classification: an application to rumour stance classification in Twitter. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 393–398, 2016.

[109] Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie J Dorr. Using paraphrases for parameter tuning in statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 120–127. Association for Computational Linguistics, 2007.

[110] Nitin Madnani, Joel Tetreault, and Martin Chodorow. Re-examining machine translation metrics for paraphrase identification. In *Proceedings of the 2012 Conference of the NAACL: Human Language Technologies*, pages 182–190. ACL, 2012.

[111] Daniel P Maki and Maynard Thompson. Mathematical models and applications: with emphasis on the social, life, and management sciences. *Mathematical Models and Applications*, 10:12, 1973.

[112] Heikki Mannila, Hannu Toivonen, and A Inkeri Verkamo. Discovery of frequent episodes in event sequences. *Data mining and Knowledge Discovery*, 1(3):259–289, 1997.

[113] Kent Marett and KD Joshi. The decision to share information and rumors: Examining the role of motivation in an online discussion forum. *Communications of the Association for Information Systems*, 24(1):4, 2009.

[114] Jon D Mcauliffe and David M Blei. Supervised topic models. In *Advances in neural information processing systems*, pages 121–128, 2008.

[115] Marcelo Mendoza, Barbara Poblete, and Carlos Castillo. Twitter under crisis: Can we trust what we rt? In *Proceedings of the first workshop on social media analytics*, pages 71–79. ACM, 2010.

[116] Jean-Baptiste Michel, Yuan Kui Shen, et al. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182, 2011.

[117] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *ArXiv Preprint ArXiv:1301.3781*, 2013.

[118] Dmitrijs Milajevs, Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Matthew Purver. Evaluating neural word representations in tensor-based compositional settings. *ArXiv Preprint ArXiv:1408.6179*, 2014.

[119] E Morozov. Swine flu: Twitter's power to misinform @ONLINE, 2009. Available: http://neteffect.foreignpolicy.com/posts/2009/04/25/swine_flu _twitters_power_to_misinform [Accessed: April 15, 2015].

[120] Lev Muchnik, Sinan Aral, and Sean J Taylor. Social influence bias: A randomized experiment. *Science*, 341(6146):647–651, 2013.

[121] Maziar Nekovee, Yamir Moreno, Ginestra Bianconi, and Matteo Marsili. Theory of rumour spreading in complex social networks. *Physica A: Statistical Mechanics and its Applications*, 374(1):457–470, 2007.

[122] Dung T Nguyen, Nam P Nguyen, and My T Thai. Sources of misinformation in online social networks: Who to suspect? In *MILCOM 2012-2012 IEEE Military Communications Conference*, pages 1–6. IEEE, 2012.

[123] Nam P Nguyen, Guanhua Yan, My T Thai, and Stephan Eidenbenz. Containment of misinformation spread in online social networks. In *Proceedings of the 4th Annual ACM Web Science Conference*, pages 213–222. ACM, 2012.

[124] Peter Norvig. Statistical learning as the ultimate agile development tool. In *CIKM*, 2008.

[125] Shigehiro Oishi, Jesse Graham, Selin Kesebir, and Iolanda Costa Galinha. Concepts of happiness across time and cultures. *Personality and Social Psychology Bulletin*, 39(5):559–577, 2013.

[126] Evelien Otte and Ronald Rousseau. Social network analysis: a powerful strategy, also for the information sciences. *Journal of Information Science*, 28(6):441–453, 2002.

[127] Par. Paraphrase identification (state of the art) @ONLINE, 2016. Available: http://aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art) [Accessed: July 15, 2016].

[128] Pavel Pecina, Antonio Toral, Vassilis Papavassiliou, Prokopis Prokopidis, Josef Van Genabith, and RIC Athena. Domain adaptation of statistical machine translation using web-crawled resources: a case study. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation*, pages 145–152, 2012.

[129] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. Wordnet::similarity: Measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, HLT-NAACL–Demonstrations '04, pages 38–41, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[130] Politifact. Politifact. `http://www.politifact.com/`, 2017. [Online; accessed 19-July-2017].

[131] S Pramod and OP Vyas. Data stream mining: a review on windowing approach. *Global Journal of Computer Science and Technology Software & Data Engineering*, 12(11):26–30, 2012.

[132] Vahed Qazvinian, Emily Rosengren, Dragomir R Radev, and Qiaozhu Mei. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599. Association for Computational Linguistics, 2011.

[133] Daniel Ramage, Susan Dumais, and Daniel Liebling. Characterizing microblogs with topic models. *International Conference on Web and Social Media*, 10:1–1, 2010.

[134] Jacob Ratkiewicz, Michael Conover, Mark Meiss, Bruno Gonçalves, Snehal Patil, Alessandro Flammini, and Filippo Menczer. Detecting and tracking the spread of astroturf memes in microblog streams. *ArXiv Preprint ArXiv:1011.3768*, 2010.

[135] Jacob Ratkiewicz, Michael Conover, Mark Meiss, Bruno Gonçalves, Snehal Patil, Alessandro Flammini, and Filippo Menczer. Truthy: mapping the spread of astroturf in microblog streams. In *Proceedings of the 20th International Conference Companion on World Wide Web*, pages 249–252. ACM, 2011.

[136] Jacob Ratkiewicz, Michael D Conover, Mark Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer Menczer. Detecting and tracking political abuse in social media. In *Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[137] Ralph L Rosnow. Inside rumor: A personal journey. *American Psychologist*, 46(5):484, 1991.

[138] Ralph L Rosnow and Eric K Foster. Rumor and gossip research. *Psychological Science Agenda*, 19(4):1–2, 2005.

[139] Richard M Ryan and Edward L Deci. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, 55(1):68, 2000.

[140] Christin Seifert, Barbara Kump, Wolfgang Kienreich, Gisela Granitzer, and Michael Granitzer. On the beauty and usability of tag clouds. In *12th International Conference on Information Visualisation.*, pages 17–25. IEEE, 2008.

[141] Eunsoo Seo, Prasant Mohapatra, and Tarek Abdelzaher. Identifying rumors and their sources in social networks. In *SPIE Defense, Security, and Sensing*. ISOP, 2012.

[142] Devavrat Shah and Tauhid Zaman. Rumors in a network: who's the culprit? *Information Theory, IEEE Transactions*, 57(8):5163–5181, 2011.

[143] Chengcheng Shao, Giovanni Luca Ciampaglia, Alessandro Flammini, and Filippo Menczer. Hoaxy: A platform for tracking online misinformation. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 745–750. International World Wide Web Conferences Steering Committee, 2016.

[144] Snopes. Snopes. `http://www.snopes.com`, 2017. [Online; accessed 19-July-2017].

[145] Stefan Stieglitz and Linh Dang-Xuan. Emotions and information diffusion in social media?sentiment of microblogs and sharing behavior. *Journal of Management Information Systems*, 29(4):217–248, 2013.

[146] Alexander Strehl, Er Strehl, Joydeep Ghosh, and Raymond Mooney. Impact of similarity measures on web-page clustering. In *Workshop on Artificial Intelligence for Web Search (AAAI 2000*, pages 58–64. AAAI, 2000.

[147] Tamara Taggart, Mary Elisabeth Grewe, Donaldson F Conserve, Catherine Gliwa, and Malika Roman Isler. Social media and hiv: a systematic review of uses of social media in hiv communication. *Journal of Medical Internet Research*, 17(11), 2015.

[148] Dennis Thom, Harald Bosch, Steffen Koch, Michael Wörner, and Thomas Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *Pacific Visualization Symposium (PacificVis), 2012 IEEE*, pages 41–48. IEEE, 2012.

[149] Salvatore Trani, Diego Ceccarelli, Claudio Lucchese, Salvatore Orlando, and Raffaele Perego. Dexter 2.0: an open source tool for semantically enriching data. In *Proceedings of the 2014 International Conference on Posters & Demonstrations Track-Volume 1272*, pages 417–420. CEUR-WS. org, 2014.

[150] Rob van der Goot and Gertjan van Noord. Rob: Using semantic meaning to recognize paraphrases. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 40–44, 2015.

[151] Maria Grazia Vigliotti and Chris Hankin. Discovery of anomalous behaviour in temporal networks. *Social Networks*, 41:18–25, 2015.

[152] Hao Wang, Dogan Can, Abe Kazemzadeh, François Bar, and Shrikanth Narayanan. A system for real-time Twitter sentiment analysis of 2012 us presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations*, pages 115–120. Association for Computational Linguistics, 2012.

[153] Kuansan Wang, Xiaolong Li, and Jianfeng Gao. Multi-style language model for web scale information retrieval. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 467–474. ACM, 2010.

[154] Kuansan Wang, Christopher Thrasher, Evelyne Viegas, Xiaolong Li, and Bo-june Paul Hsu. An overview of microsoft web n-gram corpus and applications. In *Proceedings of the NAACL HLT 2010 Demonstration Session*, pages 45–48. Association for Computational Linguistics, 2010.

[155] Weixin Wang, Hui Wang, Guozhong Dai, and Hongan Wang. Visualization of large hierarchical data by circle packing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 517–520. ACM, 2006.

[156] Wikipedia. Fake news @ONLINE, 2018. Available: https://en.wikipedia.org/wiki/Fake_news [Accessed: April 15, 2015].

[157] Yingcai Wu, Shixia Liu, Kai Yan, Mengchen Liu, and Fangzhao Wu. Opinionflow: Visual analysis of opinion diffusion on social media. *IEEE Transactions on Visualization and Computer Graphics.*, 20(12):1763–1772, 2014.

[158] Wei Xu, Chris Callison-Burch, and Bill Dolan. Semeval-2015 task 1: Paraphrase and semantic similarity in Twitter (pit). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 1–11, 2015.

[159] Wei Xu, Alan Ritter, Chris Callison-Burch, William B Dolan, and Yangfeng Ji. Extracting lexically divergent paraphrases from Twitter. *Transactions of the Association for Computational Linguistics*, 2:435–448, 2014.

[160] Wei Xu, Alan Ritter, and Ralph Grishman. Gathering and generating paraphrases from Twitter with application to normalization. In *Proceedings of the sixth workshop on building and using comparable corpora*, pages 121–128, 2013.

[161] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 927–936. ACM, 2009.

[162] Fabio Massimo Zanzotto, Marco Pennacchiotti, and Kostas Tsioutsiouliklis. Linguistic redundancy in Twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 659–669. Association for Computational Linguistics, 2011.

[163] Wenbin Zhang and Steven Skiena. Trading strategies to exploit blog and news sentiment. In *ICWSM*, 2010.

[164] Yating Zhang, Adam Jatowt, Sourav Bhowmick, and Katsumi Tanaka. Omnia mutantur, nihil interit: Connecting past with present by finding corresponding terms across time. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 645–655, 2015.

[165] Jian Zhao, Nan Cao, Zhen Wen, Yale Song, Yu-Ru Lin, and Christopher Collins. # fluxflow: Visual analysis of anomalous information spreading on social media. *IEEE Transactions on Visualization and Computer Graphics.*, 20(12):1773–1782, 2014.

[166] Yuchen Zhao and Philip Yu. On graph stream clustering with side information. In *Proceedings of the Seventh SIAM International Conference on Data Mining*, pages 139–150. SIAM, 2013.

[167] George Kingsley Zipf. *Human behavior and the principle of least effort: An introduction to human ecology*. Ravenio Books, 2016.

[168] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2):32, 2018.

[169] Arkaitz Zubiaga and Heng Ji. Tweet, but verify: epistemic study of information verification on Twitter. *Social Network Analysis and Mining*, 4(1):163, 2014.

[170] Geoffrey Zweig and Christopher JC Burges. The microsoft research sentence completion challenge. Technical report, Technical Report MSR-TR-2011-129, Microsoft, 2011.