Inferring Interestingness in Online Social Networks

Will Webberley

2013

Cardiff University
School of Computer Science & Informatics

Declaration

This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed ..... (candidate)

Date .....

Statement 1

This thesis is being submitted in partial fulfillment of the requirements for the degree of PhD.

Signed ..... (candidate)

Date .....

Statement 2

This thesis is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by explicit references.

Signed ..... (candidate)

Date .....

Statement 3

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ..... (candidate)

Date .....

To People you care
for their patience and support.

Abstract

Abstract text here etc.

Contents

List of Publications

Some of the work produced towards this thesis has also been published separately as follows.

- W. Webberley, S. M. Allen, R. M. Whitaker. *Inferring the Interesting Tweets in Your Network*, in *Workshop on Analyzing Social Media for the Benefit of Society (SOCIETY 2.0), 3rd International Conference on Social Computing and its Applications (SCA)*, Karlsruhe, Germany. *IEEE 2013*

- W. Webberley, S. Allen, R. Whitaker. *Retweeting: A Study of Message-Forwarding in Twitter*, in *Workshop on Mobile and Online Social Networks (MOSN'11), 5th International Conference on Network and System Security (NSS)*, Milan, Italy. *IEEE 2011*

List of Figures

List of Tables

List of Acronyms

OSN Online Social Network

MTW Mechanical Turk Worker

Glossary

Author

A user that has written a Tweet (*the author of Tweet* t *is denoted as* $A(t)$).

Follower

A type of user. A user, *x*, is a follower of user *y* if user *x* follows user *y*. Other users who follow a particular user will receive all of the user's Tweets and retweets to their home timeline. A user can elect to follow another user.

Friend

The inverse of follower. User *x* is a friend to user *y* if *y* follows *x*.

Path-length

The penetration of a Tweet - i.e. the number of times a Tweet is retweeted down one chain. The final retweeter in the chain indicates the number of hops the Tweet has taken from its author.

Retweet

n. - A replica of a Tweet, which has been forwarded on by a user (who is not the Tweet's original author) to their own followers (*denoted as* $RT(t)$).

v. - The act of replicating a Tweet. A user who finds a Tweet interesting may retweet it so that it gets more exposure.

Retweet Group

Set of Twitter users responsible for the propagation of a Tweet. Comprises the original

author of the Tweet and the users which have since retweeted it.

Retweet Count

The number of times a particular Tweet has been retweeted.

Timeline

A collection of Tweets in Twitter in reverse-chronological order. A **user** timeline consists of that user's Tweets. A user's **home** timeline consists of the Tweets of each friend of the user.

Tweet

n. - A piece of information in Twitter; a piece of text, less than 140 characters long, which is written by a user. When sent, the Tweet is pre-pended to its author's user timeline and also to the home timelines of each of the followers of the Tweet's author (*denoted as* t).

v. - The act of writing and sending a Tweet.

Note - A Tweet, in the context of Twitter, is treated as a proper noun and as such has its first letter capitalised[1].

User

An account on Twitter. Each user (usually representing a real-life person or organisation) can Tweet, retweet, follow other users and be followed by other users. In this thesis, sometimes the terms *user* and *person* are used interchangeably.

[1] https://twitter.com/logo

Chapter 1

Introduction

Online social networks have exploded into the lives of millions of people worldwide over the last decade, and their use has dominated the communication highways and facilitated the interconnection of the world in ways never before perceived possible.

These social networks imitate real-world social networks. Although most such platforms each provide a different service to collaboratively satisfy an array of different use-cases, they tend to all be based around the idea of 'friendships' (i.e. links between the user nodes in the social graph) and the sharing of information amongst friends.

Social networks like these have been available for around ten years now (with MySpace[1] launching in 2003 and Bebo[2] in 2005), but it wasn't really until Facebook's[3] worldwide launch in 2006 that social networks became the staple, ubiquitous norm that they are today. More recently, we have seen the introductions of Google's social network grown from its Buzz service, Google Plus[4], Pinterest[5], App.net[6], and many more. They make up a large part of the basis and meaning behind the ideas of Web 2.0, which describes the web as being primarily formed from user-generated content and encourages the sharing of such content.

Another component that helped in the dawn of Web 2.0 was the rise of *blogging*. A

[1] http://myspace.com
[2] http://bebo.com
[3] http://facebook.com
[4] http://plus.google.com
[5] http://pinterest.com
[6] http://app.net

blog ('web-log') is a time-based series of posts consisting of continuous pieces of text, photos, or other media, and is generally contributed to by a single author. Blogs are often based around one or a set of topics and are usually public - meaning that they are written with the intention of being read by others. Despite this, they are often a way in which the author can look back at their history of posts, acting more as a diary recording snapshots of the author's life.
Various blogging services exist on the web today, such as Medium[7], Wordpress[8], and Tumblr[9].

1.1 Twitter as a Social Network

Twitter[10] is an online social network, which launched in the summer of 2006 [**?**]. Since then, it has rapidly gained in popularity amongst several different user groups - teens and young people, casual users, celebrities, reporters, and so on - and within eight months had around 94,000 registered users [**?**]. Although Twitter has never been a direct competitor with Facebook, users tend to use the two sites concurrently for different purposes: whilst Facebook's focus is on providing many services at once (such as photo-sharing, commenting/endorsing of information, messaging, pages for businesses, groups, events, etc.), Twitter's is more on simplicity.

More specifically than just being an online social network, Twitter is a microblogging website. Whilst a blog, as mentioned, typically contains long posts, Twitter only allows its users to post short pieces of text, up to 140 characters in length [**?**] [**?**], called 'Tweets'. Thus, Twitter is a hybrid social network and blogging service and whilst each Tweet may only realistically be able to hold a couple of sentences, this system facilitates quick, timely, and 'real-time' *live* information-sharing amongst its millions

[7] http://medium.com
[8] http://wordpress.com
[9] http://tumblr.com
[10] http://twitter.com

of users [**?**]. Its idea is that short pieces of news will 'travel' faster and will be seen by more people more quickly than traditional news stories.

Although Tweets are limited to 140 characters in length, the inclusion of URLs is allowed. This enables further extension of Tweets through external websites, and supports the inclusion of links to images and videos. Twitter has encouraged this use-case by providing 'share' buttons for developers to embed in websites, and direct support for photo and video applications, such as TwitPic[11] and Vine[12].

Its simplicity has also helped its growth into the mobile domain, in which smartphone users are able to very quickly post updates about their lives, a piece of information they want to share, or a photo or video, and be able to post it *as it happens* directly from the news source or geographical location [**?**]. This has been especially useful in emergency situations worldwide, including the Haiti earthquake in 2010 [**?**], and 2011's Egyptian protests [**?**] and Thai flood [**?**].
Indeed, [**?**] used Twitter to build an earthquake-reporting system for Japan that outperforms the Japan Meteorological Agency in terms of its promptness of notification.

Use of Twitter is based around 'timelines' of Tweets, to which new Tweets are prepended as they are posted by users. The *home* timeline is the default view, in which Tweets from all of a person's subscribed-to users are placed. Timelines of an individual user contain only Tweets from that user, and are known as a 'user' timeline. Customisation of timelines is also possible through the use of Twitter lists, in which different users can be placed to categorise streams of Tweets from different sets of users.

[11] http://twitpic.com
[12] http://vine.com

1.2 Twitter's Social Graph and Information Subscription

As with many social networks, the structure of Twitter lies within the users and their connectivity within its social graph. However, unlike Facebook, whose social structure is made up of bi-directional 'friendships' between users, Twitter's primary social graph is made up more of mono-directional links between its users [**?**]. A person using Twitter can elect to *follow* another user, which subscribes the person to receive all of that user's Tweets to their home timeline. The set of users that follow a person are known as that person's *followers*, and the set of users that the person follows are the person's *friends*.
Therefore, if two users both mutually follow each other, then the link between them is bi-directional.

Whilst bi-directional links are common amongst communities of similar interests, friends, colleagues, and so on, mono-directional links are found more in situations in which less-influential users follow more-influential users, such as celebrities.

1.3 The Problem

A user who follows a set of other users can *generally* be said to find that set of users to produce more interesting information than those users that the user does not follow. However, despite that, not *all* information produced by an 'interesting' user is likely to be interesting, and yet *all* information produced by a Twitter friend will be received onto the home timeline.
Noise is a common problem in Twitter, and is the uninteresting information one might receive that conveys little interest. It is likely that most of the information received on Twitter *is* uninteresting [**?**], and this makes it very hard to distinguish the interesting information from the uninteresting.

Since people tend to use Twitter most in short sporadic moments, looking for a quick news fix, they do not have time to filter out noisy information. Thus, the presence of noise can dampen the experience of the user, making it much more difficult to find interesting information.

In addition, Twitter users typically exist within an information 'bubble'. This is similar to the notion of the Google search bubble, in which the search engine uses previous results and search terms to only return information to a user based on what *it thinks* the user would find the most interesting and useful.
This results in the users not knowing which information exists beyond the confines of their bubble, and if they do not know it exists, they cannot know if it is of interest to them. Similarly, a Twitter user cannot follow all of the users he/she may find interesting, since he/she will not *know* of all the interesting users existing on the social graph.

How can users be exposed to *interesting* and *relevant* information, but without them having to know about it or look for it first?

1.4 Contributions

This thesis focuses on understanding information propagation, and how this combined with knowledge of the social structure of Twitter can assist towards solving the problem of identifying interesting and relevant information and determining it from the noise on Twitter. Whilst other work in the area has also looked into the notions of relevance and interest in online social networks, and Twitter in particular, none has addressed the problem in such a way as this.

Part of the outcome of this research are methods for effectively inferring interesting information and, indeed, ranking information by interestingness. The methods are validated in various ways to help highlight their strengths and weaknesses in performing inferences and appropriate use-cases.

The work addresses the problem area in that it helps towards solving the goal of identifying *globally* interesting information in Twitter. In addition, certain measures are taken in an attempt to address the idea of information relevance, which denotes how information interestingness is subjective, and thus different from user to user.

1.5 Thesis Structure

The rest of this thesis is structured as follows.

A background is provided as an introduction to some of the ideas behind the main research, which immediately follows this chapter, and includes a review of relevant literature across the range of topics addressed in the thesis.
Following this are chapters that contain research on Twitter's information propagation characteristics and its interesting and useful behaviours, the social structure of Twitter and the ways in which this is important for understanding the spread of information, and then on the research of the methodologies themselves, including validation and analysis of the results of this work.

The thesis ends with a general analysis and conclusion, and a discussion of potential future work in this area and leading on from this research.

Chapter 2

Background

One of the most widely-used features of Twitter is its inbuilt function for facilitating the spread of information within its social structure. This phenomenon is the basis for much of the research in this thesis and, when combined with the characteristics of Twitter's user graph, has many interesting attributes and behaviours associated with it.

2.1 Domain Context

2.1.1 Information Propagation through Retweeting

The function of propagation in Twitter is known as *retweeting*, and is carried out by the Twitter users themselves. When a user views a Tweet that they believe to be particularly interesting, and believe it to also be interesting to his/her followers, then he/she can elect to retweet it, and thus pass it further through the social graph to that user's followers also. A Tweet that has been retweeted is known as a *retweet*, and it is clear that a Tweet which is retweeted will be made available to significantly more users than a Tweet that isn't retweeted [**?**] [**?**].

Since Twitter's social graph is decentralised and retweeting occurs between individual groups of users, its properties are similar to information dissemination in other types of decentralised graphs, such as content-forwarding in opportunistic networking [**?**].

A retweet can be carried out in one of two ways: either through the use of Twitter's native retweet button, or manually.

The retweet button is displayed along with each Tweet in a Tweet timeline which, when clicked, immediately creates a new retweet containing the verbatim content of the original Tweet and automatically sends it on to the retweeter's followers.

The user who created the original Tweet is credited as the author on the recipients' timelines, with an indication of who carried out the retweet itself. Thus, users who follow the retweeter will see a Tweet appear in their home timeline from someone that they may not directly follow.

Figure 2.1: ***A retweeted Tweet.***

The manual approach involves physically copying the content of the Tweet to be retweeted and pasting it into a new Tweet, usually with the text '`RT @<username>:`' pre-pended, where `RT` stands for **re**tweet and `<username>` is the username of the author of the original Tweet. This method allows for annotating the original content of the Tweet (for example, to provide an opinion on the Tweet contents), producing a *modified* Tweet, which can sometimes be pre-pended with `MT` rather than `RT`.

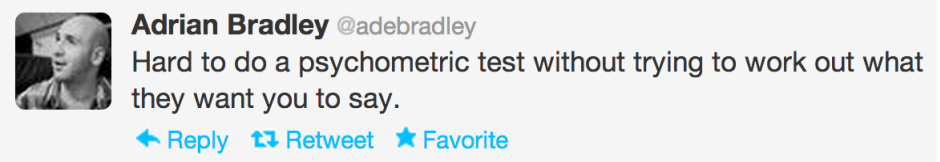

Figure 2.2: ***The retweet 'button' in context.***

Each Tweet has a retweet count associated with it, which is the raw representation of the number of times that the Tweet has been retweeted using the retweet button method.

Since the manual retweet technique is more community-driven, there is no official way to include these as part of the retweet count of the original Tweet. However, since the manual method is typically only really used with the aim to annotate or modify the Tweet in some way, the resultant 'retweet' is no longer a real representation of the content of the original Tweet anyway, and so should not be counted as such.

It should be noted that Twitter users may choose to make their account 'protected'. A person who has a protected account will still have a publicly-visible profile (displaying a name, username, bio, and so on), but their Tweets and other information (such as the followers and friends lists) are hidden from users that aren't followers of the person. Potential followers of a protected account must *request* a followship, which can then be accepted or rejected by the protected account holder.
Since Tweets from a protected account are only visible to approved followers, the retweet button is unavailable for them to disseminate the Tweet any further than the author's immediate local follower network. However, since the manual retweet method does not rely on the button and isn't governed by Twitter, a protected account's Tweets can still be retweeted in this way.

In a similar way to Facebook supporting the endorsement of information found on its site by inviting users to 'like' a piece of content, retweeting is effectively a *vote* or endorsement for a Tweet on Twitter. In both cases, the number of likes and number of retweets is visible to the platforms' respective users, and so this provides some insight into the *popularity* of the information.

2.1.2 Retweets and the Social Graph

The social graph of Twitter is constructed, like in other online social networks, by edges between users, partially emulating real-life social interactions between humans. The growth of social media has encouraged more dense communication between users

all over the world, who would not previously be able to be in direct contact with one another in this way.

Derived from this, Stanley Milgram's finding of "six degrees of separation" [**?**], which defines that people are usually no more than six hops away from each other on the 'real-life' social graph, was found to be an overestimate when it comes to the analysis of the structure of OSNs by [**?**], who found that the average 'distance' observed in Facebook's entire 721 million-node graph in 2011 was only around 4.7 hops. This implies that denser links between users and larger communities that apparently manifest themselves in OSNs create a smaller 'world' than that experienced in reality.

In each of Milgram's experiments participants passed a message to one another, at each stage only passing to other people that they actually *know*, in the hope of the it reaching a single intended recipient. This meant that people could use acquaintances in other geographic locations to transfer the message from community to community.
Twitter supports a similar propagation mechanism in the fact that retweets can themselves be retweeted; this is a focus of some of the earlier research in this thesis.

This behaviour provides further penetrative 'depth' of the information through the social network away from the source user in addition to the spread 'width' made by the initial retweets. Although retweeting is not carried out with the aim of information reaching any particular final user (or set of), as with Milgram's experiment, this phenomenon allows retweets to 'travel' between 'online communities' of users.

As with real-life social networks, communities of users in OSNs are also a common feature [**?**].
In Twitter, these communities are typically small to begin with and are based on a topic of interest or around a more influential user. As more Tweets are produced from within the community, further links are made to interconnect the community's users, producing a growing 'swarm' of interest around the initial topic or user [**?**].
As further users begin associating themselves with this community, its audience becomes more widespread and the community grows. This concept is discussed in greater

length by [**?**], who also experiment further with communities and describe them as compact groups of users connected by dense follower links.

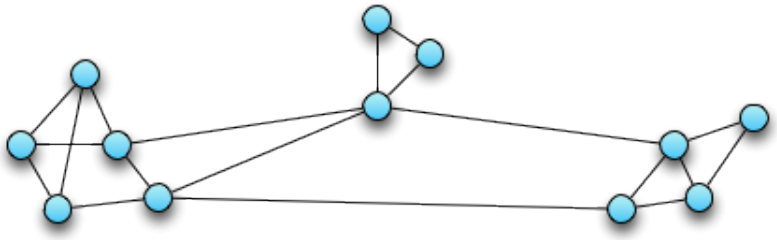

Figure 2.3: ***A hypothetical group of user communities.***

In more dense communities, Tweets can be made available to many users immediately after they are published, since many of the links between users are shared. This means that any retweets that occur within communities are likely to have a lot of *redundancy*, in that many of the retweets will be sent to users who have already seen the Tweet. Although Twitter prevents this information duplication by not showing the retweets of Tweets that have already appeared on a user's timeline, it does increase the chance of the Tweet making its way out of the community.

Retweets amongst users within a community are likely to be common, due to the shared-interest nature of communities, and some users can provide 'bridges' by being active in more than one community. In these cases, Tweets can be passed between the communities through retweets by the bridging user. If there are many users sharing communities, then there are many more avenues available for propagation to occur down, causing a high level of information throughput. If there are fewer bridges, then there is more of a bottleneck between the communities, hindering the information spread.

[**?**] also finds that communities can be formed from different types of people, such as those who Tweet frequently and have many followers, and those who contribute very little and have few followers. Those with many followers and many friends receive lots of information and have the potential to spread information further than those with

fewer inward and outward edges. Studies in the behaviour of different types of users in Twitter is done more thoroughly in [**?**], which defines 'broadcasters' (users with many followers and few friends) and 'miscreants' (users with few followers but many friends) and their roles in information propagation.

Users that retweet the interesting information from a source user to others, who do not follow the source user and so would not naturally receive the information, are effectively acting as information *filters*. By not following the source user, a person might still receive the interesting information through these filters, but will not receive any of the 'noise'. Thus retweeting means that friends of a user become useful filters of information for users further 'downstream' and retweeted information can be said to have a higher *credibility* than Tweets that aren't retweeted [**?**].

2.1.3 User Influence

Just as there are different types of user *behaviours* on Twitter, as mentioned in the previous section, the are also users of different *influence* levels [**?**].

Much research has gone into user influence, including on how this might be detected [**?**], and influential users are generally found to be those that have a greater impact on Twitter's social network [**?**] and that usually have significantly more followers than an average user. Influential users tend to have a high persuasion over other users, relating *influtentials* in Twitter to those who are also influential in the real world as part of traditional communication theory [**?**], and therefore many Twitter influentials are the accounts belonging to real-world celebrities.

As with real-world celebrities, Twitter influentials are those with many 'influenced' followers, or fans, which are the users who have the strongest agreeable opinions of the influential. As a result, an influential user has a greater number of followers who are interested in the information produced by the user, and is therefore more likely to receive more retweets than less influential users.

Although influence level is partly derived from the follower count of the user, it should be noted that a user with high in-degree on the social graph[1] does not necessarily imply a high level of influence. An 'active' audience of users who reply, retweet, and interact are more indicative of an influential user [**?**]. This is especially true since a user can gain more followers through campaigns such as '#teamfollowback'[2] or by following 'out of politeness', in which a user will follow another user back as an act of politeness, but these users tend to have *both* high in- and out-degree and invoke less interactivity amongst their followers, which are not necessarily characteristics of an influential user [**?**].

Klout[3] is a web service that attempts to review a user's social media influence by assigning users a Klout Score. Their website declares that this score, which ranges from 0 to a maximum of 100 and whose generation algorithm is kept private and unpublished [**?**], is determined from a variety of 400 sources taken from eight different social media platforms, and which *also* seems to take interactivity between users as the primary indicator [**?**]. Additionally, the service indicates the topics a user is influential about, with the general idea being for organisations to check up on which users are influential for marketing purposes, but also to highlight the users that should be replied-to at a higher priority.

2.1.4 Twitter as an Information Retrieval System

From a high level, Twitter is essentially just a variety of information-retrieval system, which people can utilise to produce and consume information when required. In traditional information-retrieval systems, such as search engines and library systems, keywords and search terms are common ways for describing the type of information the user would like to receive back. The system would then search a database or archive

[1] In-degree: many followers

[2] Users associate themselves with #teamfollowback to imply they will return all followships.

[3] http://klout.com

for what it believes is relevant information, *based* on these 'retrieval parameters', and return results to the user ordered usually by the estimated relevance of the articles [**?**].

Information quality is also reliant on the expected reading effort of the returned documents. The character precision-recall metric was introduced by [**?**] by way of demonstrating the tolerance-to-irrelevance ratio. The general mechanism for this ratio is to do with users reading a document passage; the point at which this ratio is reached is when the user stops reading the particular passage and moves to the next whole document, since they assume the rest of the document is also irrelevant to them.

Therefore, the more effective the information retrieval system is in displaying high-quality information, the lower the chance that this ratio is reached by the user.
It is comparable that a Twitter user viewing Tweets from a user they are following may get to the point where he or she reaches this ratio (i.e. is beginning to get bored or find the Tweets irrelevant) and decides to unfollow the friend. Similarly, the more effective the user is when selecting people to follow in the hope of receiving interesting information, the less likely it is that the user will remove these friends.

Whilst Twitter does not support the use of keyword searching for its primary information delivery method, it does lend its users some control over the type of information they wish to receive. As mentioned previously, users receive all of the Tweets from everyone that they follow onto their home timelines. Thus, by selecting users to follow, a person is effectively describing and implicitly indicating the type of information he/she would like to receive, and by editing their friends list (either by adding new followers or pruning existing ones) he/she can alter this indication.

Despite this control, it is still unlikely that users will achieve a perfect Twitter experience due to the presence of *noise* [**?**]. As discussed in the Introduction, this problem stems from that although a person follows users they consider to be interesting, it is often the case that not *all* information produced by interesting users will be interesting itself.

2.1.5 Information Quality, Popularity and 'Interestingness'

Information-retrieval systems typically use some measure of information *quality* when determining which documents to return to a user and also when deciding on the *order* the documents should be displayed in. This 'quality' is subjective in that different systems use a variety of different algorithms for deducing quality, usually based on the level of *interest* in each of the available documents (such as Google's Page Rank algorithm and Amazon's recommendation algorithms), but also in that the level of quality itself depends on the user itself requesting the information.

In the case of Google's Page-Rank, the algorithm uses multiple cues to determine who the user is, their interests, past searching habits, links clicked, and so on, to return *relevant* information, which is incidentally one of the causes of the aforementioned Google search bubble.
Amazon's recommendation algorithms analyse a user's past item views and purchases and cross-matches these against trends based from users who also looked or bought similar items. Amazon is then able to accurately determine the type of items a customer are interested in purchasing, and can send emails to that customer with personalised recommendations.

Thus, information quality is essentially a function of information interestingness and information relevance, which are both related to the concept of *effective stimulation* [**?**] discussed later.

Twitter uses no such metrics to deliver information to its users, relying on the users themselves to implicitly 'choose' the information they want to receive - it is an information retrieval system and not a recommendation system. Additionally, information is always displayed in chronologically-ordered timelines, with new Tweets being continuously inserted at the top as they occur. Twitter does not try to indicate interesting Tweets on the timeline which means that the interesting information is shown at equal value alongside the 'noisy' Tweets, causing the difficulties in identifying the interest-

ing information as has been mentioned previously.
Indeed, the recent TechCrunch article from October 2013, "Twitter Quitters And The Unfiltered Feed Problem"[4] talks at more length about this particular phenomenon, and helps highlight the problem area of this work more clearly.

The retweet count of a given Tweet is a useful metric in inferring a Tweet's *popularity*. If a Tweet is retweeted 10 times, then ten people have taken the time to read that Tweet, decide it is worth sharing, and then actually retweet it [**?**]. This user (and the other nine retweeters) may have found the Tweet interesting, yet it should be noted that although the count can be used as a measure of popularity, as a function of the influence of the Tweet's author, the retweet count alone cannot be used as a measure of how interesting the Tweet actually is [**?**]. For example, it is inappropriate to say that the first Tweet in Figure 2.4 is so significantly more *interesting* than the second, although it is clearly more popular since Justin Bieber is an extremely influential Twitter user.

Figure 2.4: ***Example of Tweets with significantly different retweet counts.***

Whilst the work in this thesis does not aim to build an accurate retweet-predictor, this does become a basis for some of the work in later chapters.
[**?**] identifies the same problem of 'noisy' Twitter timelines and discusses methods for predicting *popular* Tweets using a J48 decision tree classifier, based on the likelihood of the Tweet being retweeted by a particular user. Although the authors address information relevance from a user-centric point of view, the validations of whether a

[4] http://techcrunch.com/2013/10/05/sorry-my-feed-is-full

prediction of a retweet occurring for a given Tweet is actually indicative of the *interestingness* of said Tweet do not perform particularly well.
A retweet-prediction model based on a factor graph model is introduced by [**?**] to determine how retweetable a Tweet is on a global scale. A precision of just under 29% is achieved in predicting if a Tweet will be retweeted, but no mention is made of how this relates to how *interesting* the information is.
Another study into retweet prediction was carried out by [**?**], in which a trained probabilistic collaborative filter model (named 'Matchbox') was used to determine the useful features in making the predictions. As with the previous study, the research focuses on a retweet *probability*, which is a binary decision made by one particular user. The methodology is not aimed at the inference of interestingness, and simply determines that the most relevant features for accurate decision predictions are the author of the original Tweet and the retweeter.

Inversely, [**?**] and [**?**] predict the *type* of messages that are likely to be retweeted further, the latter using a logistic regression to both predict an individual retweet decision and a retweet *volume*. The methods do not apply these notions to how interesting the information actually is, achieve low recall and the multi-classifications seems only to perform well on very unpopular or very popular Tweets. It is made clear, however, that the retweet volume of a Tweet is useful in denoting Tweet *popularity*.

[**?**] uses a passive-aggressive machine-learning algorithm to make binary predictions on retweet decisions and cited that social features - for example, number of followers of the author, frequency of Tweeting, etc. - were the largest factors in the performance, and [**?**] uses a logistic regression, partly using a dataset published as part of another paper by the same authors as [**?**], to predict retweet decisions in order to address information interestingness. However, little effort is made to define interestingness or, indeed, validate that the inferences towards this are accurate and correct.
A logistic regression is again used by [**?**] for predicting binary retweet behaviours with the focus on information propagation in disaster scenarios, and [**?**] showed that condi-

tional random fields can perform better than logistic regressions than when modelling retweet behaviour in the same way.

Since the above papers only effectively consider a prediction of retweet outcome, which is a binary decision, it is hard to relate this to more of a global interestingness, aside from stating that a retweet implies the retweeter's relative interest in the Tweet. However, a retweet count, as mentioned above, is inappropriate as an indicator of *magnitude* of interest, and so the research into predicting individual retweet decisions cannot be used as a basis for this. Additionally, not much emphasis is placed on how well the techniques work 'on-the-fly'; many of the methodologies discussed require several features that may take a long time to collect and compute, making them unsuitable for use as part of quick and useful interestingness evaluations.

The idea of Tweet scoring and retweet *count* predictions is introduced by [**?**], who used their methodologies to produce a system[5] enabling users to compile Tweets in ways that are predicted to achieve the most retweets. The predictions are based on averaging the score, derived through a linear regression, of different components of a user's Tweets (such as the inclusion of a particular hashtag), so that when a Tweet by the same author is next constructed, the various components of the new Tweet can be compared against the scores of the counterparts seen in previous Tweets. The value produced through this method is then used to generate an expected retweet count as part of a comparison to the user's average ('baseline') achieved retweet count at this point in time, and was shown to perform well on influential Twitter users.

However, the methods described do not take into account fluctuations in the social graph, particularly in the case of less-influential Twitter users, who's local networks are prone to more frequent changes. Additionally, they rely on enough previous Tweet and temporal information on the user to be evaluated, and do not relate the resultant score to any type of interestingness metric in the context of highlighting it from amongst noise.

[5] https://sites.google.com/site/learningtweetvalue/home

Alonso et al. ([**?**]) also use 'scoring' to address interestingness, focusing more on determining *uninteresting* content, by assigning Tweets an integer score out of five. Although the authors initially attempted to train a decision tree classifier on a set of 14 features, they settled on classifying a Tweet as 'possibly interesting' if it simply contains a URL, and otherwise classify it as 'not interesting'. Although the authors did then further classify the possibly interesting Tweets, by studying the magnitude of the crowdsourcees used to evaluate the Tweets that found them interesting, and then classifying Tweets based on them containing a particular type of named entity - for example, a person's name, a place or brand name, and so on - the categorisation system is too coarse and is not capable of representing the many different types of Tweets seen on Twitter.
Additionally, despite achieving relatively high accuracy in this particular area, the methods are not suitable for assessing Tweets on a general or user-specific level, especially since Tweets that don't contain URLs might still contain interesting content.

An interesting study is described by [**?**], in which a clustering algorithm is used, taking into account the retweet count of a Tweet and how this is related to the popularity of the source user, to determine information quality. Although this work is more similar to the research discussed later in this thesis than others, the scoring is quite simple and the author's use-case seems limited to that of identifying the most important Tweets surrounding a particular event (such as the death of Michael Jackson).
Additionally, the authors do not make any effort to verify their results in any way, aside from comparing the Tweets determined to have a high quality by each of their two assessed methodologies.

2.1.6 Precision and Recall

Precision and recall are two metrics that are often used simultaneously to verify the performance of a method or procedure, with the usual goal being to maximise both. The metrics are used for validating *accuracy* in different ways, yet they can be applied

to other purposes also and are useful in describing the notion of interestingness in Twitter.

The precision and recall measures are talked about somewhat in Twitter- and retweet-based literature. These pieces tend to only analyse the measures on their own work when applied to Twitter rather than on any more global scale. Certainly, there is less in the literature on the subjects of precision and recall with regards to retweeting in general.

The idea of assessing the credibility of information is introduced in [**?**], in which the authors demonstrate methods of measuring the credibility of 'news' and 'chat' Tweets. In this case, retweeting is seen as a possible measure of a Tweet's credibility, since users typically only retweet information they see as interesting or useful. The authors use a logistic regression on a set of features derived from each Tweet in order to classify its credibility.

The precision and recall metrics are used to verify the different aspects of the paper's results. In particular, they are applied to the classification of assessing credible information (and users) in order to calculate how well classified the information is. A higher precision, therefore, shows that their model has accurately classified most of the total information classified as either credible or non-credible.

$$Precision = \frac{\text{Number of correct classifications}}{\text{Number of total classifications made}}$$

$$Recall = \frac{\text{Number of correct classifications}}{\text{Total number of potential classifications}}$$

On a similar note, [**?**] discusses the notions of precision and recall more generally. The authors discuss the problem regarding the balance of information received by Twitter users. Having too few friends reduces the number or interesting posts received (i.e. low recall); having too many friends may cause information overload and is likely to include a lot of noise (i.e. low precision). This issue is used, instead of to validate

results, as a basis for the work; predicting the Tweets that are most popular and will be retweeted the most.

In addition, precision and recall are used to compare the method to two other baselines; the TF-IDF score and *Retweet Before*, which uses the fact that if a Tweet in the training data has been previously retweeted, then it's likely to be retweeted again. The two metrics are also used to compare results when certain features are removed from the classifier. For example, showing that without using a 'user retweet' feature, the precision and recall remain significantly higher than when removing other features, meaning that this feature does not contribute highly to the performance.
More specifically, precision and recall are used in a similar way to in [**?**]; except rather than looking at the number of classifications made, the authors use the number of predicted retweets.

[**?**] discusses a proof of concept for detecting influential users in one of two categories; evangelists or detractors. Precision and recall, in this case, are used slightly differently:

$$Precision = \frac{\text{Number of influential users retrieved}}{\text{Number of users retrieved}}$$

$$Recall = \frac{\text{Number of influential users retrieved}}{\text{Total number of users}}$$

The concept is taken further through the use of another metric, the *Mean Average Precision*, which is used to denote an influential user as being a detractor or an evangelist. A high precision, in this case, would imply a large proportion of influential users are classified correctly and a high recall means that most of the influential users existing in the entire dataset have been classified. The final results then show the precision and recall values for detecting evangelists and detractors in both follower/following networks and interaction networks. Both precision and recall improved when the size of the set of highest classified influentials increased (i.e. the top set of influential users).

[**?**] presents a method for the automatic classification of Twitter information to determine if a document is positive, negative or neutral in sentiment. In this case, the authors

replace precision with *accuracy* and recall with *decision*, since they are using many classes instead of a binary classification, and define them as the following:

$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Number of all classifications}}$$

$$Decision = \frac{\text{Number of retrieved documents}}{\text{Number of all documents}}$$

The accuracy is measured across the classifier's decision, and the $F_{0.5} - measure$ is then calculated based on these values instead in order to show that the classifier works well when the dataset size is increased.

As well as a good news source, Twitter is also used as an informational, user-contributed source on world events. [**?**] introduces a system, TwitInfo, which can be used for detecting, summarising and visualising events from Tweets. The authors looked at football match footage, web content, and earthquake survey data, and manually annotated major events in each to produce ground truth sets. These would be use to compare and contrast the results produced by their event detector using the following definitions of precision and recall:

$$Precision = \frac{\text{Number of events detected were from ground truth set}}{\text{Total number of events}}$$

$$Recall = \frac{\text{Number of events detected}}{\text{Number of events in ground truth set}}$$

With these definitions set, the authors were then able to easily calculate precision and recall for their algorithm.

For the work in this thesis, interestingness of information is the performance metric used to describe information quality, and thus precision and recall for any particular user in the scope of this thesis can be defined as follows:

$$Precision = \frac{\text{Number of interesting Tweets received}}{\text{Total number of Tweets received}}$$

$$Recall = \frac{\text{Number of interesting Tweets received}}{\text{Total number of all interesting Tweets}}$$

where *received* means that the Tweet has arrived on the user's home timeline, but does not imply that the user has *read* the Tweet.

Therefore, a user following many other users will receive lots of interesting information onto their home timeline in amongst lots of noise; resulting in a reduced precision and higher recall. Another user might follow a very select few other users who are of direct interest, and thus will experience high precision, but low recall.
These metrics are therefore useful in describing the concepts of noise and interestingness, and are consistent with their respective definitions in that users will achieve an optimum Twitter experience if both precision and recall are maximised.

Zadeh et al. ([**?**]) defined bespoke definitions of precision and recall, yet also in the domain of interesting information on Twitter. Although the authors identify the need for users to be able to discover other users of interest and declare that Twitter does, in fact, have a 'high precision' of interesting information, they admit to using a very coarse set of possible interest categories and is only based on *overlapping* interests rather than addressing the interest-noise ratio more concerning the research in this thesis. Additionally, clicks on URLs by users are the only means by which to measure this interestingness, and Tweets with URLs are usually the most interesting type of information [**?**].

2.2 Collecting Twitter Data

Most of the analytical work in this thesis relies on various data being collected from Twitter. Twitter provides an API for developers in order to facilitate the production of applications for its platform, but also for research purposes. It permits interfacing with many components of Twitter's service, such as posting and retrieving Tweets, interacting with other users (e.g. creating new friendships), and most of the features

that Twitter's service itself provides to its users.
The API encourages use of the OAuth[6] authorisation framework to handle access[7], allowing Twitter to keep track of applications and each application's access privileges and rate limits[8].

Twitter's traditional REST API, v1[9], provided many useful endpoints for data collection and allowed each OAuth-authenticated application 350 hourly POST and GET requests[10].
In June 2013 Twitter officially deprecated v1 of its REST API, forcing use of its new v1.1 API[11]. The new version contains many of the same resources[12] as the original, but workarounds are required to get the results as some of the endpoint requests possible through v1. Additionally, new rate-limit policies were introduced, allowing more limited and controlled access to most of the available resources.

Since the work in this thesis was ongoing over this switch-over date, the initial work utilised API v1, and the latter work API v1.1, causing some changes to some of the data-collection methodologies as the thesis progresses. Descriptions of the data-collection in each relevant part of the thesis reflect this change, where appropriate.

2.3 Research Motivation

The motivation for the work in this thesis lies in the need to distinguish interesting information from noisy Tweets in Twitter, the latter of which is the problem area identified over the previous sections of this thesis.
It has been made clear that the retweet count of a Tweet cannot reliably be used as a

[6] http://oauth.net
[7] https://dev.twitter.com/docs/auth
[8] https://dev.twitter.com/docs/rate-limiting/1.1
[9] https://dev.twitter.com/docs/api/1
[10] https://dev.twitter.com/docs/rate-limiting/1
[11] https://dev.twitter.com/blog/api-v1-retirement-date-extended-to-june-11
[12] https://dev.twitter.com/docs/api/1.1

measure of interestingness, especially in the context of influential users, who naturally achieve significantly more retweets than average users, but which does not imply that the information they produce is of a higher quality or interest level.

As a result, the retweet count alone cannot be useful in distinguishing interesting information from noise in a timeline of mixed Tweets from different users with different levels of influence - some further metric is required to make this distinction.

This thesis covers the procedure and research behind a methodology that determines and ranks information on Twitter through inferences of interestingness that allows the more interesting information to be brought forward.

Chapter 3

Understanding The Behaviour of Retweeting in Twitter

Stuff to finish up in this section:

- Explain motivation for research in this particular area
- Use this motivation to explain the purpose for this research as a basis for the work in the next few chapters
- Explain how this chapter is the basis for research into Twitter's social structure in the next chapter
- Normalise terms (retweet-group size / retweet volume) here and in further chapters throughout thesis
- (thinking forward: e.g. we have addressed tweet quality in terms of propagation, can a network have a quality too? what further factors can affect the dissemination of information in social networks?

It has been discussed that the popularity of information in Twitter can be related to the propagation characteristics of that information through Twitter's social structure. That is to say, that the more times a Tweet is retweeted by users, the more people have found the information contained within it to be interesting enough to be worth sharing.
It has also been shown that this retweet count metric alone cannot be an implication of

the actual interestingness level of a Tweet. This is related to the notion of user influence, which directs that some tweets are naturally immediately seen by more people and thus have a higher chance of achieving a retweet as they are. Indeed, since follower count is one of the [**?**] demonstrated that a user's Tweets' retweet rates increase as the user's follower count increases.

The strength of Twitter lies is in its social structure, where users can elect to follow and unfollow others as they choose. Followers of a user receive all of that user's posts in their individual (or 'home') timelines. If a user has set their profile to be public, then their posts also used to appear on the public timeline, which is now deprecated but was accessible to anyone; even those without a Twitter account. As a result, people are likely to follow users who update with interesting posts; whether the follower is a big fan of the user and simply wants to know everything going on in their life, or if the follower is simply interested in the topical area of most of the friend's posts.

Just as Twitter users will post Tweet about topics that are of interest them - possibly related to a user's work, a hobby, or a mixture of multiple areas - and these Tweets are generally posted with the idea that they will be useful or interesting for some of their followers as well as an attempt to attract more followers, retweets are generated with the same motives in mind. This means that if a Tweet is retweeted, it is not only allowed to disseminate further through the social structure, but also that a higher Tweet quality is implied.

Thus, this describes how a user's friends, who carry out retweets, effectively become filters of interesting information for that user and other followers of those friends, and the *audience* of the original Tweet is significantly increased. Since retweets are always attributed to the original author then you, a Twitter user, may gain more attention by means of followers by posting *interesting* Tweets, which will;

1. increase the chances that users reading your Tweets will choose to follow you, and

2. increase the chances that users will decide to retweet your Tweet, thus broadcasting it to a larger audience. People viewing this *retweet* then may decide to follow you.

Since a Tweet can be retweeted multiple times, and, as mentioned, a retweet itself can also be retweeted, the much larger the effective audience (both directly and through retweets) of a Tweet's original author has the potential to become if they choose to post interesting information. In this chapter, an understanding of the behaviours and properties of retweets is provided, along with discussions into how these are relevant in determining useful metrics for determining which retweeted information is interesting.

3.1 Tweet and Retweet Properties

3.1.1 Retweet Groups

A Tweet has various attributes associated with it, which make up the features that describe that particular Tweet. Each Tweet has a set of properties relating to its content, its author, and other metadata, such as creation time.
As such, a particular Tweet, t, can have its relevant properties declared and be defined as follows;

$$t = (\text{text}, \text{count}_R, \text{author}_O, \text{author}_R, \text{orig})$$

Respectively, this represents the Tweet's text, its retweet count, and the *original* author of the Tweet. The final two values depend on whether t is a retweet or not and represent the author of the retweet and the original Tweet respectively. Since a retweet remains a class of Tweet, then the same properties can be assigned to retweets as to Tweets, except that in the case of retweets the values orig and author_R will be non-null.

Since a Tweet can be retweeted more than once, the set of Tweets that are in the set of

all Tweets, T, and are retweets of t is defined as;

$$RT(t) = \{s \in T : s.\text{orig} = t\}$$

Clearly, the retweet count of t is $t.\text{count}_R = |RT(t)|$.

An original Tweet, t, along with all of the retweets of t, $RT(t)$, are known as the *retweet group* of t, which is defined as $G(t)$ and is useful when discussing the audience reach of a particular Tweet. Therefore, since t is also a member of this set, the size of t's retweet group is;

$$|G(t)| = t.\text{count}_R + 1$$

Which can have a minimum size of two - the original author and at least one retweeter. If $r_1, ..., r_n$ are the members of $RT(t)$ then the raw audience size of the group can be calculated thus (assuming $t.\text{count}_R \geq 1$);

$$\text{audience}(G(t)) = \text{followers}(t.\text{author}_O) + \sum_{i=1}^{t.\text{count}_R} \text{followers}(r_i.\text{author}_O)$$

However, properties of Twitter dictate that this raw audience size is not an accurate calculation in most cases, as is discussed later in this chapter.

3.1.2 Retweet Trees

As a Tweet gains in popularity and attracts more and more retweets to be created from it, and since retweets themselves can also be retweeted, then this ultimately results in the generation of a retweet *tree*, which represents the retweet group of a particular Tweet. This tree is formed from the *users* who have retweeted the Tweet (or a retweet of the Tweet), and represents the original Tweeter and the various pathways taken by the Tweet as it is retweeted through the social graph.

[**?**] also uses retweet trees to assist in illustrating information dissemination in Twitter, particularly in observing the Twitter reactions to the 2009 Air France airline crash.

The tree is not a representation of the actual social ties between the tree's nodes, as users are able to retweet Tweets and retweets sent from others that they do not follow. However, as is mentioned later in this chapter, most retweeting does generally occur between directly-linked users.

The root of the tree representing every $G(t)$ is $t.\text{author}_O$ and, if t has been retweeted, each of the other nodes are

$$r_1.\text{author}_R, ..., r_n.\text{author}_R \, \forall \, 1 \leq n \leq t.\text{count}_R$$

A similar illustrative device is used by [**?**] in describing URL *cascades* in Twitter.

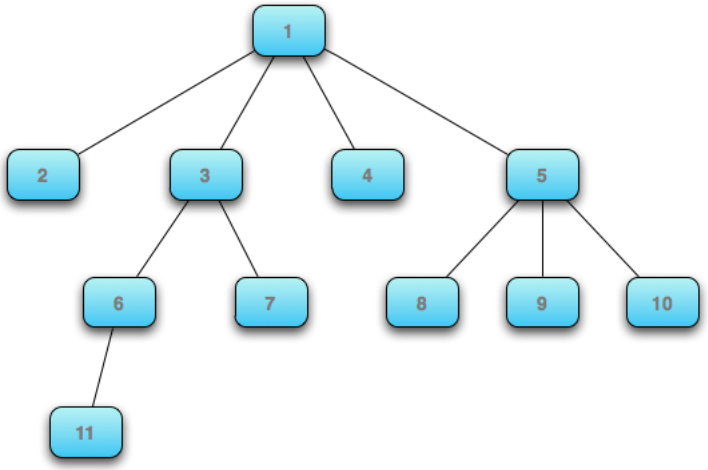

Figure 3.1: ***A hypothetical retweet pathway tree.***

Although these retweet pathways can technically be acyclic through the use of the manual retweet method, the case of a user retweeting a Tweet more than once is very rare and a user retweeting a retweet that they are already part of the upstream chain of is even less likely to occur. The retweet button method simply does not support users retweeting a Tweet more than once.

As such, retweet trees are used in preference over retweet *graphs* as they illustrate the temporal nature in terms of the order in which the retweets occur.

3.1.3 Path-Length

In addition to retweet groups having a size property, a retweet groups's branch's *path-length* refers to the length of a particular retweet chain. In particular, it defines the number of times a Tweet is retweeted down one chain from the source user (the retweet group's tree's root) down to the final retweeter in the chain (a tree's leaf node).

Figure 3.1 represents the users in the retweet group of a hypothetical Tweet.
This retweet group has a size of 11 and has 7 distinct retweet chains, the longest of which is the one traversing users 1, 3, 6 and 11.
The *maximum* path-length of this retweet group is therefore 3, as the leaf node of this branch is three hops away from the original author at the root.

As has been mentioned previously, when a user retweets a Tweet or retweet through the manual approach, it involves pre-pending the current state of the Tweet with the text `RT @<username>:`.
Therefore, the Tweet with the content;
`RT @user2: RT @user1: This is the body of the Tweet`
was originally authored by `user1`, then retweeted by `user2`, and then finally retweeted by the author of this current retweet (the author of a Tweet or retweet's username is not credited in the body of the text).

It should be noted that this phenomenon can only be observed through retweets by the manual approach, since the button method always simply credits the original author, and not any of the internal members of the retweet group.

Although most retweets today are carried out using the button method, the manual approach still remained popular at the time the research in this chapter was carried out. This allowed for making useful observations of retweet patterns that would not be as prevalent later on.

3.2 More on Information Retrieval

A Twitter user electing to follow another user cannot predict precisely what the new friend will Tweet about in the future. The user has some *expectation* of the type of information they are likely to receive based on the previous Tweets of the new friend, which is generally the only cue the user can use to base the follow decision on.

Part of the follow decision is based on the notion of relevance judgement, which is a notion discussed at more length by [**?**] and is partly made up of the goal of achieving *affective stimulation* through *hedonic* searching as opposed to the use of *epistemic* searching.

3.2.1 Epistemic Search

An epistemic information search is one that involves carrying out a search with the purpose of finding out information on a particular topic (or set of) to satisfy a *desire for knowledge* [**?**], yet without an actual aim to solve any particular problem.

An example of this type of search is a 'crawl' through Wikipedia, in which a searcher may start at one particular page of interest and then follow links within that page to other related pages of interest that stem away from the source topic. In this case, the search 'parameter' is simply the name or title of the article the searcher wants to view. As mentioned previously, a followship between users is effectively a search parameter in Twitter, since the following user has elected to follow the new friend to receive information from him/her. It is clear that this type of 'searching' cannot be epistemic as the following user cannot know exactly the type of information they are going to receive.

3.2.2 Hedonic Search and Affective Stimulation

Hedonic searching is similar to epistemic searching in that it is also not carried out with the aim to solve an immediate problem, but is different in that it is done to search for fun or 'affective stimulation' [**?**].

A person can be said to be affectively stimulated if they view a piece of information that has some effect on the person, such as an emotional effect, something that is of particular interest to the person, or something that is capable of provoking some further thought.

With hedonic searching, users are not aware of the information that they are going to receive prior to searching and thus cannot really predict any level of affective stimulation.
This aligns more with Twitter usage, since users receive information that they cannot accurately predict. Any Tweets received that do and provide interesting information convey affective stimulation to the user. This is the type of information that becomes harder to identify amongst lots of noise, yet is also the type of information a user is more likely to retweet.

3.2.3 The Recognition Heuristic

A further metric for measuring information relevance in information retrieval is the recognition heuristic.
The recognition heuristic takes advantage of a person's memory and declares that if a person is able to recognise only one of two (or more) items, then he/she is more likely to judge the recosnided item to be 'greater' or more important [**?**] [**?**].

Relating this to information received on Twitter, [**?**] found that a user recognising a Tweet's author significantly increases the chance that the user will decide to read the Tweet. Since a user must read a Tweet in order to make a decision on whether, or not,

to retweet it, then the recognition heuristic transiently plays a part in a user's retweet decision also.

The authors also find that information about the Tweet itself, such as its text content and its retweet count, has much more of an effect on a user's read decision than information about the author, such as the followers count or Tweet rate. This also contributes to the declaration that information interest goes beyond the features surrounding a particular user and that user influence does not dictate interestingness of information.

3.3 Twitter Propagation Analysis

Understanding information propagation in Twitter is the key to also understanding how interesting information might be detected. Whilst it is known that the retweet count of a Tweet cannot be used alone in inferring interestingness, since this is simply a level of popularity tied in with the author user's influence, it is still a factor in that users are more likely to retweet interesting information than noise.

Of particular interest is to achieve an overview of propagation behaviours in Twitter, the patterns in the properties of retweet groups, such as their sizes and penetration depth, temporal aspects of retweets and information on the social structure of Twitter itself with regards to propagation within it.

The remainder of this chapter involves an exploratory study of the retweet characteristics in Twitter to provide a further background, and which demonstrates the area's relevance towards the goal of inferring interesting information.

3.4 Retweet and Retweet Group Analysis

To assist in providing a further grounding in this area of research, a series of analyses were carried out into retweets and retweet groups. This section describes the processes

and purpose of the analyses.

3.4.1 Data Collection Methodology

The analyses involve the examination of Tweets extracted from Twitter. Twitter's REST API v1 was used between 26th and 24th to collect around 26,000 Tweets, which represent a total of around 4,400 retweet groups. The complete set is made up of three subsets, the use of each individually is described later.
The relatively limited size of the dataset is acknowledged, yet it should be emphasised that these analyses are simply exploratory and are not used to answer or solve any specific problem.

The data collection involved a mixture of using Twitter's timelines and its search capabilities. Version 1 of the REST API supported retrieval of Tweets, 20 at a time, from the Twitter *public* timeline. Historically, this timeline contained the 20 most recent Tweets published by all the authors that have non-protected Twitter accounts and used to be visible on their website's homepage[1] to non-logged-in users.

In particular, the public timeline endpoint was queried periodically to retrieve the current set of most recent public Tweets. From all of the retrieved Tweets, the Tweets that were retweets were filtered out and stored.
Retweets, as mentioned earlier, are distinguishable since they start with the characters 'RT' followed by a username. It should be noted that when retrieving Tweets from Twitter's APIs that even retweets that were created using the button method begin with the same character sequence, allowing detection of these also.

Following storage, the text of the retweets were parsed in order to extract the text that the original Tweet contained. Sometimes, retweets using the manual approach are used to provide additional annotation to the Tweet. Although usually this can be distinguished by the fact that the original Tweet is inside quotation marks (“ ”), this

[1] http://twitter.com

is not true in all cases, meaning that sometimes the original text could not be reliably extracted programmatically by a machine.
In these cases additional queries were made to Twitter's search API in an attempt to resolve the problem, yet, failing that, the retweet was discarded.

Once the original text had been successfully extracted, this was used along with other metadata as query parameters to Twitter's search API in order to try and find the original Tweet and any other retweets of this Tweet. The search API uses approximate (or 'fuzzy') string matching, but quotation marks can be used to retrieve search results based on an exact string pattern[2].

Once the API search was complete (in some cases, with Tweets achieving many retweets, many API calls were required in order to page through results), the original Tweet could easily be identified as the only one of the set *not* starting with the sequence "RT". This provided a retweet group comprising the original Tweet and all available retweets of this Tweet.
On some occasions, more than one Tweet were each identified as the original Tweet and so the entire set was discarded. This could occur, for example, if many users may Tweet exactly the same text if it comes external sources, such as a news webpage, and means that the entire set of retrieved Tweets are not likely to be part of the same retweet group. In cases where no results were returned, the retweet was discarded and assumed an orphan retweet (perhaps as a result of a retweet of a Tweet posted by a protected Twitter account). And in cases where no original Tweet could be identified, it was sometimes possible to calculate it through cross-matching against other retweets in the retrieved retweet group, but if not they, too, were discarded.

The retweet groups were finally stored along with relevant metadata in order to carry out the studies described in the following sections.

[2]https://dev.twitter.com/docs/using-search

3.4.2 Exploring Retweet Group Path-Lengths

The path-lengths of each chain in a retweet group can be calculated by identifying the users involved in retweeting down that chain; from the original author to the final retweeter. The *maximum* path-length of a particular retweet group is the longest path-length observed in the group.

Identification of path-lengths can be carried out through parsing the text of a retweet, and following the citations. Although it cannot be guaranteed that all users will be properly cited in a chain, and there is no realistic method to verify this, it is felt that correct citations will be made enough times to make these cases relatively insignificant.

On average, the maximum path-length observed across the retweet groups was around 1.8, with the vast majority of retweet chains being between one and two edges in length. When one considers that many retweets are made through the button method, which removes citations of internal users in the chain and simply credits the original author, which would produce many single-length retweet chains, this average could theoretically be an underestimate.
[**?**]'s similar obsevations in the area also indicate a large number of groups with maximum path-lengths of one and two.

The longest observed maximum path-length was nine, which is a huge depth of penetration through the social structure since the total number of users involved in propagating the Tweet was ten. This, combined with the knowledge that social networks can represent a more tightly-knit social graph than the real world's six degrees of separation (see Introduction), shows how retweeting can have a huge impact in information spread amongst millions of people very quickly.

Also of interest is the relationship in terms of the social ties between different the different user members of a retweet group.
In cases where a retweet group's maximum path-length is precisely one, i.e. the situation where a user (or set of) has retweeted a particular Tweet only once, the retweeters

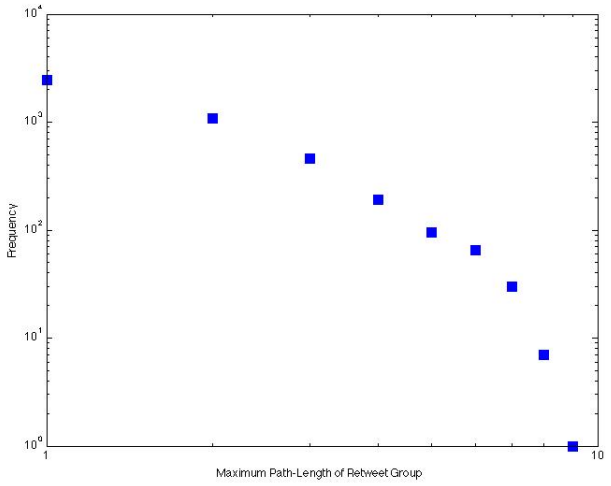

Figure 3.2: ***Log/log distribution of maximum path-lengths observed across retweet groups..***

at the leaves of this group's retweet tree follow the original author around 90% of the time.

This implies, therefore, that in the remaining 10% of cases, a retweeter has retweeted a Tweet from outside of their home timeline and has instead seen a Tweet whilst browsing through another user, who isn't a friend, timeline that the retweeter regards as sufficiently interesting.

This helps to demonstrate that the more followers a particular user has, the greater the chance that another user somewhere has of viewing the user's Tweets and then having the opportunity to retweet them. The fact that 90% of retweets of a particular user are created by direct followers reinforces this further.

This particular property could also be due to use of the button method of retweeting, which does not cite intermediate retweeters, and thus always imply that the final retweeter directly retweeted the Tweet from the original author. However, there may, in fact, have been other retweeters in between the final retweeters and original author, each of which following the immediately upstream retweeter.

As such, this 90% follow probability between the retweeter and source user in 1-hop

retweet chains is also likely to be an underestimate.

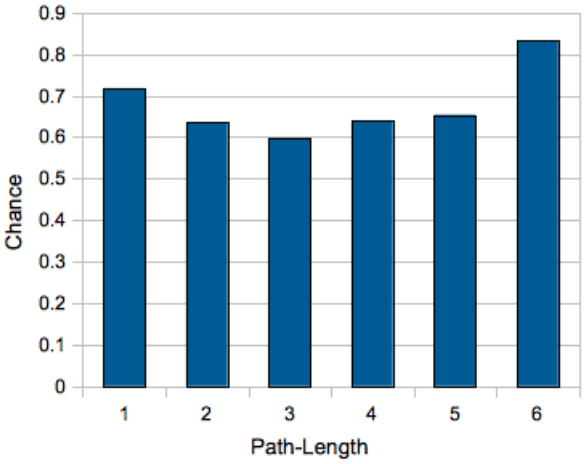

Figure 3.3: ***Proportion of cases where the original author is cited with varying maximum path-length of retweet group..***

Further to this, in situations in which the maximum path-length of a retweet group is *greater* than one, the final retweeter follows the author of the original Tweet about 40% of the time. It is clear from Figure 3.5 that retweet groups with a longer maximum path-length tend to have a larger size themselves. This increases the likelihood that the Tweet has been able to spread both further around the original Tweet's author's community, but also the potential for the Tweet to 'travel' to other communities.
Since users from outside the source user's community are less likely to follow the source user, this explains the reduction in the followship likelihood between further downstream retweeters in the retweet chains and the original author.

3.4.3 Size of Retweet Groups

The distribution of $|G(t)|$ across all of the original Tweets $t \in T$ collected from Twitter was found to follow a power-law type distribution, with a relatively large p-value of around 0.87. 3.4 represents the complementary distribution function demonstrating the changing probability of a randomly generated X being greater than or equal to x, the

'current' value of $|G(t)|$, at each stage.

The techniques used in this analysis are adapted from the methods and code provided by [**?**].

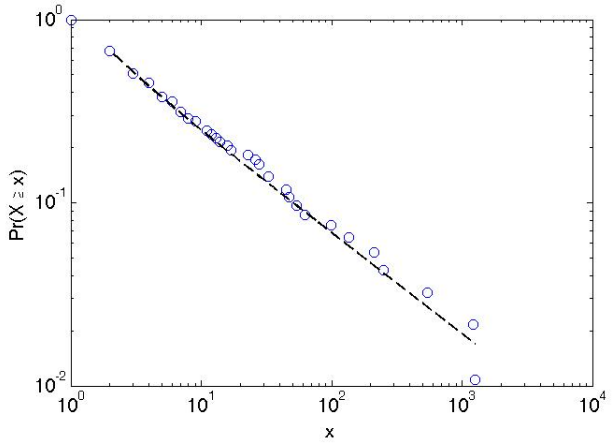

Figure 3.4: ***Maximum likelihood power-law fit for the cumulative distribution of retweet group sizes..***

The mean group size from this dataset was found to be just below six, and the largest size was 284. The smallest $|G(t)|$ were the cases in which $t.\text{count}_R = 1$, and which were significantly the most common occurrences.

Of interest also is the relationship between a group's size and its maximum path-length. Generally, the maximum path-length of a group, $G(t)$, increases with $|G(t)|$, indicating a mostly uniform growth in the retweet trees representing these groups - as might be expected. Thus this illustrates that as the retweet count of t increases, then the longer the retweet chains in $G(t)$ are likely become. This would increase its penetrative dissemination away from the source and further facilitate its spread between communities, increasing its potential *audience size*.

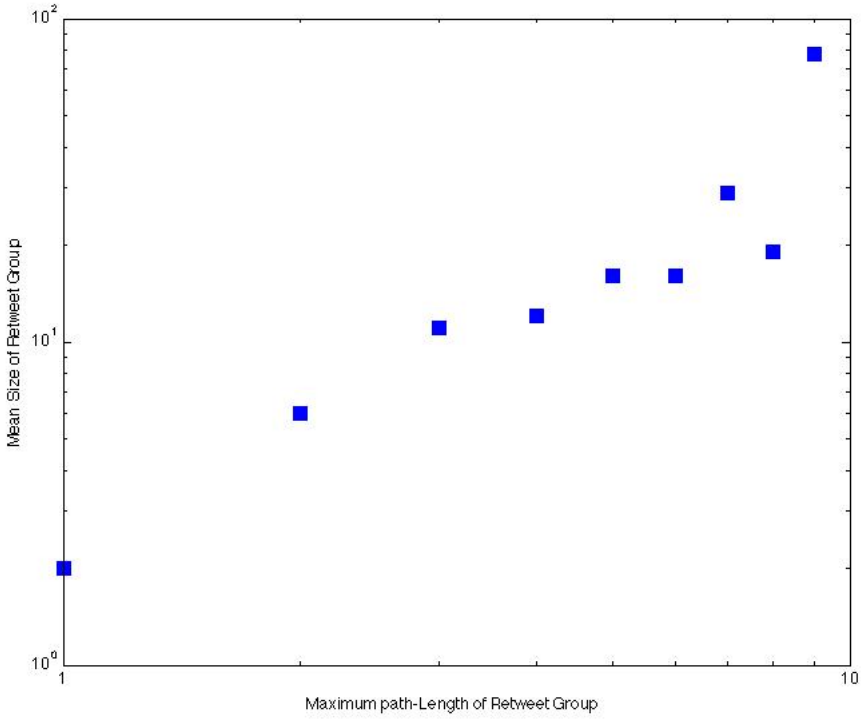

Figure 3.5: ***Log/log relationship between the maximum path-length and size of a retweet group..***

3.4.4 A Tweet's Audience - How Many Users Can be Reached?

$G(t)$'s (immediate) audience size refers to the number of Twitter users that have received t, either in its original form or as a retweet, r, such that $r.\text{orig} = t$, onto their home timelines. The term 'immediate' is used to signify the distinction between those users who passively receive the Tweet and those who see the Tweet whilst actively browsing through other user timelines or the public timeline.

Users in the latter group are therefore not direct followers of $t.\text{author}_O$ or $r.\text{author}_R \forall r \in RT(t)$ and thus cannot be tracked as members of t's audience, which, as discussed earlier, can have its size calculated through the summation of the followers of the original author and each retweeter of t.

However, this audience calculation is naÃ¯ve in that, particularly in the case of more

tightly-knit communities, users who are authors of t or $r \in RT(t)$ are likely to share a subset of each of their followers. The more dense the communities, the more followers are likely to be shared between the authors in $G(t)$ and, as such, the aforementioned audience size calculation is likely to be an overestimate in nearly all cases.
The following analyses of retweet group audience sizes relies on a dataset which began collecting at a later date than the general set used in this chapter, and thus the data represented in the rest of this section contains 2860 of the total 4400 groups originally collected. The longest maximum path-length of retweet groups observed in this subset was eight.

The *overhead* of a group, $G(t)$, which attempts to address this problem, is related to the redundency in the audience and thus represents the number of cases in which a user receives a retweet that they have already previously received the original Tweet or retweet thereof - the number of times users receive a Tweet they've already seen. The overhead also takes into account users who might receive versions of the same Tweet many times.

This overhead was found to exist in 71% of all observed retweet groups, further reinforcing that retweets often occur within communities containing users sharing links with other users. The *proportionate* overhead is the ratio of the overhead to the *distinct* audience size, which is the absolute number of users who have recieved the Tweet (or a retweet of) to their home timeline. It should be noted that the audience size does not signify the number of users who have *read* the Tweet - rather the number of users who have the *opportunity* to read it.

Effectively, therefore the distinct audience size of a Tweet can be found by modifying the earlier calculation:

$$\text{distinct audience}(G(t)) = \text{followers}(t.\text{author}_O) + \sum_{i=1}^{t.\text{count}_R} \text{followers}(r_i.\text{author}_O) - \text{overhead}(G(t))$$

Where the overhead of the group is simply the magnitude of shared followers of all authors in the group.

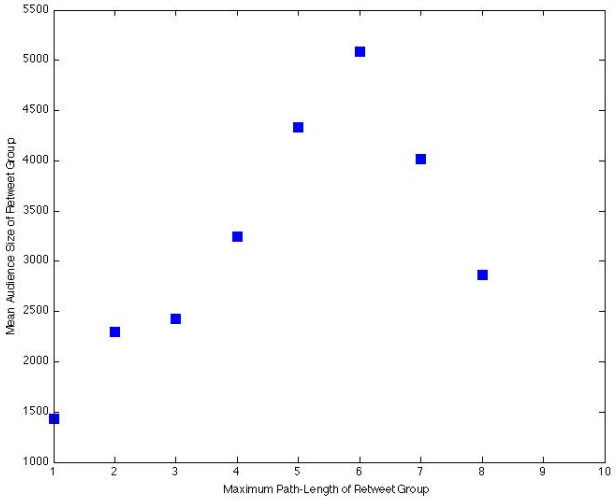

Figure 3.6: ***Relationship between a retweet group's distinct audience size and its longest path-length..***

Figure 3.6 illustrates, initially, that which might be expected; that the distinct audience size of a Tweet, t, is mostly proportional to the maximum path length of $G(t)$. However, as the maximum path-length of retweet groups exceeds 5, then a *decline* in the distinct audience size is observed. This particular behaviour has an unclear cause, but it is felt that this could be to do with a saturation in the proportionate overhead's ratio at this stage - in particular, that retweet groups attracting many retweets are circulated more within communities than outside and between communuties.
At this stage, the overhead becomes so large, causing this reduction in audience size. This is significant in that the distribution of the non-distict over the increasing path-lengths demonstrates, mostly, a continuous positive correlation.

Three of the largest five overheads in the set occur in retweet groups which have a maximum path-length of one. The *largest* overhead was of a size over six times greater than the group's distinct audience size itself, demonstrating a massive overlap between the followers of the author of the Tweet and the authors of its retweets. Whilst the audience overhead was only found to be greater than the distinct audience size in around 3% of observed retweet groups, it is still clear that the potential for overlap in the followers of

retweet group members can be very large in more closely-knit communities - i.e. those groups whose representative trees are wide and shallow.
Groups having trees with longer path-lengths typically have a proportinately lower overhead, and the chance of achieving zero overhead increases as the retweet group size decreases.

The power of the retweet phenomeon in terms of how it affects the potential audience reach of a particular Tweet is discussed in further detail by the authors of [**?**], in which they find that a retweeted Tweet of sufficient interest can reach a very large number of users even if the original author has only a few followers. The same paper more specifically mentions that the audience size of a retweeted Tweet reaches, on average, at least 1,000 users, no matter the number of followers of the original author.
This is also clear in the results in this thesis, in that even Tweets with a short maximum path-length can still have a relatively large audience size.

3.4.5 Retweet Follower Pattern

Now that an understanding has been achieved in the behaviours and properties of retweets and retweet groups, it is important that the social ties between users in groups is studied. This will provide a grounding for the research in the following chapter, in which the social structure and its role in facilitating propagation, are discussed in more detail.

It has already been shown that

The first result shown from the experiment is that the final retweeter follows the previous retweeter in the chain in 67% of cases. It initially seems strange that this should be 20% lower than when following a user in retweet chains of length one. This suggests that users involved in shorter-chain path-length retweets are members of more tightly-knit communities. Retweets with longer path-lengths have, by nature, travelled further and so would be the type of retweet to travel between communities, reducing

the chance of the involved users following each other.

The interesting part of this, however, is the number of followers of the previous retweeter in different cases. In the 33% of cases where the final retweeter doesn't follow the previous retweeter, the latter has, on average, around 600 followers. When the final retweeter *does* follow the previous retweeter, however, the previous retweeter's average number of followers is 940. This is quite a substantial difference and certainly highlights the fact that by having more followers you are more likely to have more influence in terms of whether you get retweeted, or not.

This is accentuated further when looking at the original tweeter. The likelihood of a retweeter following the original tweeter in cases in which the path-length is of more than one has already been found to be around 40%, but the average number of followers of the original tweeter increases by a factor of around four (580 to 2000) when also followed by the final retweeter. Results showed that the original tweeter had a consistently higher number of followers when followed by the final retweeter than when not at all path-lengths. This demonstrates that having an increased number of followers is correlated with the chance of a user being retweeted. In this case, having four times the followers increases the correlation dramatically (40% to 90%). The number of followers of a user can therefore be directly related to the ideas of influence discussed in [**?**] and also of 'advertising' themselves.

It was found, however, that the number of followers of the original tweeter diminishes as the path-length of the tweet increases (Figure 3.7), signifying that tweets travel further when the original tweeter has fewer followers. Because the retweet groups were collected in such a way so that groups containing longer path-length retweets also contained many shorter-chain retweets, retweet groups containing path-lengths of 5 (or more) are also likely to contain many retweets (if not more) with path-lengths of one or two (see the distribution in Figure 3.2). It can therefore be argued that there are more users involved in shorter-chain retweets than in ones with longer path-lengths. It is then more likely for these users to have more followers than others in the retweet group. Another explanation could be that users are actually aware of their local net-

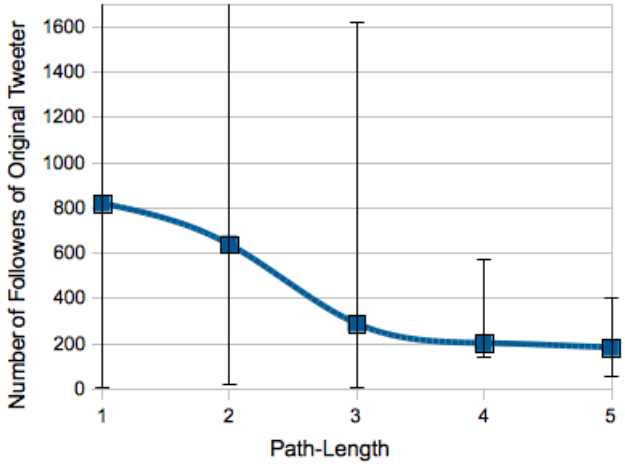

Figure 3.7: ***Relationship between number of followers (and respective distribution) of the original tweeter as the path-length increases..***

work and realise that retweeting may cause a lot of audience overlap (particularly in the case of large communities). A user may have seen a post retweeted a few times on their home timeline and thus decide not to also retweet.

One last interesting point to make regarding the notion of retweet chains is looking at the how the pattern of following previous retweeters develops as the path-length increases. It has already been discussed above how the chance of following the previous retweeter in the chain is about 67%, but, in cases where the path-length of a tweet is greater than two (i.e. at least two intermediate retweeters between final retweeter and original tweeter), the chance of the final retweeter following the next retweeter along preceding the previous retweeter is around 45%. This suggests that retweeting is more widespread and not so much just circulated around communities. These preliminary results demonstrate that the chance of the final retweeter following previous retweeters - up to and including the original tweeter - diminishes along the chain or as the tree is ascended (Figure 3.8).

Because of this, it's sensible to assume that the tweets in the dataset are forwarded through less-connected users, and perhaps forwarded from community to community by those users belonging to several groups. Otherwise, if the retweets were circulated more around closely-knit communities, the likelihood of the final retweeter following

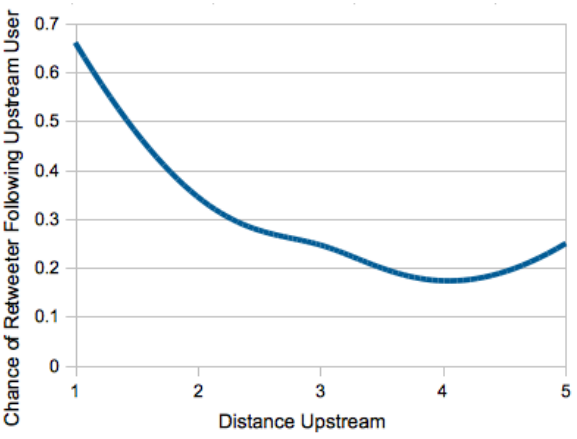

Figure 3.8: ***Proportion of final retweeters following upstream users at varying distances along the chain..***

the previous tweeters would be both greater and more evenly spread - i.e. the chance of following the previous retweeter would be roughly equal to the chance of following the other tweeters in the chain.

In addition, of the 67% of final retweeters who *are* following the previous retweeter, about 19% of them also follow the next previous retweeter (i.e. the retweeter at path-length - 2). In this case, the next previous retweeter has, on average, 3000 followers. In the 81% of these users *not* following the next previous retweeter, then the latter has an average of 525 followers. This is an accentuated result of the one previously, but this time boasts an increase of a factor of 6.

Of the 33% of users who *don't* follow the previous retweeter, about 30% follow the next previous retweeter. Both of these sets of statistics also go towards the idea of the diminishing chance of following the users as the tree is ascended.

From this dataset, it was also possible to work out how often retweeters cited the original tweeter of a post. In retweets, users are typically cited by, as we have seen, having their name along with an 'RT' at the start of the post. This data was collected by seeing if the original poster's username was mentioned *anywhere* in each retweet. The chance of this occurring was found to be around 68% and did not vary with any pattern with path-length.

3.4.6 Retweet Time Delay

The final experiments in this section focus on the time delay between the final retweeter and original tweeter. This is an interesting area since it enables researchers to see how fast messages propagate through the Twittersphere. From this information, and by using the retweeter patterns demonstrated above, it would be possible to work out how far and how quickly information can be passed around.

Figure 3.9 shows the average time delay between the first and final retweet with increasing maximum path-length of the retweet group.

The results indicate that, mostly, as the group's maximum path-length increases, then

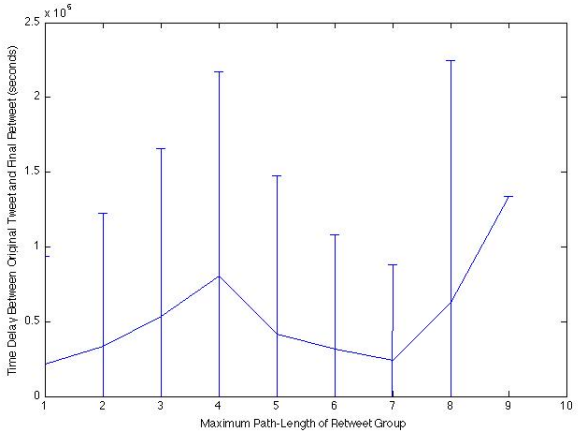

Figure 3.9: ***Average time (in seconds) between first post and final retweet of a retweet group varying with the group's maximum path-length..***

so does the elapsed time between the original post and the final retweet. This is probably as was expected, since this shows that it takes longer for a retweet to travel further. The data is not consistent, however, especially results for a path-length of five and above. The first four results suggest a uniform incline roughly proportional to $v = \frac{s}{t}$, where the distance, s, is the hypothetical distance given by the path-lengths, showing that the *speed* of propagation remains mostly constant.

There is not enough of a trend in the data to make any deductions regarding propagation speed, however. There are two main conflicting arguments regarding this result: the

first is, as mentioned, that the further a tweet travels the longer time it travels for. The second is to do with tweet popularity: the more popular a tweet is, the more quickly it will be retweeted. In the latter case, it is possible that longer trees grow fully before shorter ones, implying exponential growth. Generally, though, it seems that the maximum path-length of a retweet group does not massively affect the tweet's propagation speed.

3.5 Summary

The experimental results have certainly highlighted the ideas of communities and that of message cascading similar to that demonstrated in [**?**] and [**?**] respectively.
As has been seen, in section 3.4.6, the retweet tree seems to grow in a variety of ways. One argument is mostly expected; that as the 'distance' the tweet travels increases, then so does the time taken for it to reach its end. The other argument is linked to the idea of tweet advertising, discussed in previous sections, and namely the notion of tweet popularity.
The previous experiments showed how the number of followers of a user directly influences their chance of being retweeted. It can therefore be seen that advertising can also be linked to the level of influence a user has. The results help illustrate the multi-dimensional properties of the retweet tree, and how these factors relate to its growth and its associated friend-follower graph.

This paper has demonstrated relatively simple results in an attempt to realise some of the behavioural patterns of retweets, both linking to the psychologically in terms of the users, but also the physical properties of retweets. The results are able to represent a basis for potential further work and research into the various aspects of Twitter, moreover, perhaps, topical categorisation and the dynamicity of the friend-follower graphs.

Chapter 4

Analysis of Twitter's Social Structure

Stuff to add to this section:

- Change tweet features for each simulation and make comparison on these differences
- Observe differences in patterns when network generation parameters are altered
- Link up section to previous section (i.e. how did the previous research help and how does this build on that work?)
- Explain how this section becomes the basis for work in 'main chapter 3'.
- (e.g. Issues with current method (too long, requires network, inaccurate due to having to choose users with fewer followers), so need a quicker, more accessible and online approach).
- Explain the Mechanical Turk questions in more detail, with examples.
- Discuss about the machine learning approach used (logistic regression and how it works)
- Link 'retweet volume' to 'retweet group size'

Twitter is often seen as one of the biggest sources of new and live information on the Internet, with millions of people producing and absorbing information daily. Users

receive tweets onto their timelines from the users that they follow. Thus, a user has some control over the *type* of information they receive by choosing which other users to follow. A particular user, therefore, may not be aware of information that exists outside of their local network, since he or she is not directly exposed to the information produced by non-followees.

Retweeting allows users to forward information they receive onto their own followers. As a result, followers of these users, who might not normally be exposed to this information, now have a chance to access it. A user who decides to retweet a tweet can be said to consider that tweet to be *interesting* (at least, to their followers), since that user has taken the time to read the tweet, decided whether or not to share it, and then to actually retweet it [**?**].

Several factors can affect a user's decision to retweet, such as whether the tweet contains an (interesting) URL, whether the tweet mentions another user, whether a user even has a chance to see the tweet, the influence of the author, and so on. These factors account for a user's individual retweet *decision* on a particular tweet, and the combination of several users' retweet decisions dictate how far the tweet will propagate. However, it is our belief that the social network structure also has an affect on how far tweets can travel.

The social structure of Twitter is built up by users electing to follow other users. When a user follows a user, a directional link is forged between them, and any tweets generated, or forwarded, by the followed user are passed down the link. It's clear to see that, as more links are made between users, many more avenues are generated for message propagation throughout the social structure. Users with a high in- and out-degree can become an information highway, but users with a low out-degree are a bottleneck of information.

In this chapter, we demonstrate how different network types support different propagation characteristics through the use of a model simulating each network type. Using the model, we make predictions on the retweet outcomes on several network types, and compare these to the characteristics of the propagation in real Twitter networks. We

finish by discussing how the *interestingness* of a tweet may be inferred from simulating the network in this way.

The notion of time decay and how this is associated with a user's retweet decision is discussed in addition to a retweet probability prediction in [**?**] (and [**?**]), which is the basis of the model we use in this chapter.

4.1 Overview

In the next sections, we introduce and briefly explain the regression model we use for simulations and predictions. We then go on to analyse the differences in the propagation characteristics between three different network types before comparing the results of simulations on these networks with data collected from the Twitter social graph. We finish by introducing a methodology for predicting the interestingness of a particular tweet to a particular user and how this might be improved.
Ideally, there are two things we'd like to see from the first set of experiments; firstly, that changing the network type and properties does, indeed, affect the propagation behaviour, and, secondly, that at least some of the results from the experimentation correspond to Twitter's own retweet behaviour so that a fair comparison and justifications of our results can be made.
With regard to our prediction work, we'd like to be able to make relatively decent predictions on which tweets are of interest, and which are not, based on the simulation research in the next section.

4.2 Model

As mentioned above, [**?**] introduced and discussed their prediction model, which was shown to perform well when predicting the retweet decision of a user. Their model trains a logistic regression using a set of user-, tweet- and context- features in order

to classify an experimental tweet and output a retweet probability based on a set of features.

4.2.1 Machine Learning

Machine learning techniques are useful for making predictions based on certain input criteria based on the perceived history of previous outputs from the same inputs.

General Overview

For example, consider the three attributes, A, B and C, each of which is of a boolean data type. A machine learning technique is 'shown' that in every case where A is True and B is False, then C is true; and that in each case where A is False and B is True, then C becomes False. The history of these inputs suggests that A is strongly associated with C (the strength of this relationship will increase if the input/output history is larger), such that if the technique now tries to predict C based on the fact that A = True, then it will likely suggest that C is also True with high confidence.

A technique that has been shown many of these input/output combinations (known as 'instances') is said to be a trained model, where the model type is that of the learning technique used. This model can then make predictions based on the same inputs it has been trained on, and will therefore not work for attribute inputs that it hasn't been trained with.
Generally, in large enough datasets, the training data will not exclusively contain instances where A is the inverse of B, and so the model will be able to make predictions in cases where A = B (though if these cases are less prevalent then the confidence of these prediction outputs will be weaker). When training the model, instances must contain the input attributes as well as the value of the output attribute. When testing against the model after training, the model is supplied the input attributes and predicts

the value of the output attribute. Generally, the model will also indicate how confident it feels that the prediction is correct, and thus an idea can be obtained of the value of each of the input attributes. Some model types (particularly regressions) output instead the *probability* with which the attributes align to the trained regression.

In this example the focus has been on boolean data types, but most machine learning techniques are not limited to these. Indeed, most applications require the learning of real numbers (integers, floats, etc.) and more highly-dimensioned nominal attributes, where there are several categories that the value of one of the input attributes can lie in. Many machine learning techniques exist and some are more accurate than others when it comes to different data types and several support the notion of 'weighting' attributes, in which a strength weight is assigned to each attribute to signify the confidence with which the model should rely on that attribute for producing the prediction.
For example, for purely nominal values, then logistic regressions can be accurate in predicting outcomes, whereas for a mixture of real and nominal attributes a Bayesian attempt might be more suitable.

Logistic Regression

The work in this chapter utilises a logistic regression for outputting retweet predictions based on the feature input criteria discussed later.
As mentioned, the logistic regression predicts the statistical likelihood that the input features align with what the model has been trained with. The work in this chapter relies on the use of a retweet probability of a given user acting on a given Tweet at a certain time. Thus, it is trained in such a way so that the binary input attributes produce a binary output retweet outcome attribute: 1 for retweet, 0 for no retweet.
When testing, the binary features of the test set are used to output the probability that these features will result in a retweet.

4.2.2 Algorithm

Include image indicating the flow of events that take place for the model

Pseudocode for the algorithm

In essence, the model requires a network of users and one tweet to start the simulation. It starts by initialising a user set, *U*, to contain one source user from the network, U_s. This source user then transmits a tweet which is then received by its followers. U_s is then removed from U. For each follower, its tweet and user features are classified to produce a retweet probability. If this is greater than a random number, R, then the user retweets the tweet to its followers and the process repeats. Any user that retweets the tweet is removed from U and added to the retweet decision set RT. The followers of the retweeter are added to U for the next iteration, in which each user in U has received the tweet onto their timeline. The retweet volume of this tweet in this network is then the cardinality of the set RT.

Twitter Timelines

A tweet in a user's timeline will slip further down in the timeline as time goes by. This happens whether the tweet is interesting or not and whether or not the user has even seen the tweet. Users having a quick browse through Twitter may not have time to scroll down to find these interesting tweets (and will not know they exist) and thus tweets left in the timeline have their retweet chance decay over time.

We emulate this phenomenon in our simulations by removing the tweet from a user's timeline (by removing the user from the set U) if the user has not retweeted the tweet within a timestep threshold, which can be varied to alter the volatility of the retweet.

4.2.3 Features

We use this model as part of a simulator in order to obtain an approximate retweet volume of each tweet when the retweet probabilities of each user receiving the tweet

are combined. [**?**] used a set of around 50 features, though they mention how some features have more weight than others. In order to simplify the simulation significantly and to make data collection more tractable, we decided to use only the following four major features;

- *follows* - Whether or not the user exposed to the tweet is a follower of the author;
- *followed* - Whether or not the user exposed to the tweet is followed by the author;
- *mentioned* - Whether or not the user exposed to the tweet is mentioned in the tweet;
- *URL* - Whether or not the tweet contains a URL;

Where *author* is defined as the user who originally tweeted the tweet.

4.3 Training the Model

4.3.1 Data Collection

To collect the training data, we crawled Twitter, using the REST API, from March-June 2012 to collect a set of around 12,000 tweets and retweets from the Twitter public timeline. In addition, we made further calls to collect the information required for the features (namely the *following* and *followed* features).

4.3.2 Feature Extraction and Regression Training

FILL THIS IN

The above features were extracted in each case and the regression was trained.

4.4 Network Analyses

In this section, we look at three different network structures and discuss the differences in the propagation patterns produced by each. Note that each graph we assess is *directed*. The same set of tweets are used for each simulation on each of the networks.

4.4.1 Path Network

A path network is the most simple of the three, and is also the least life-like when compared to Twitter's own social graph. In this network, the output retweet volume is, by definition, equal to the penetration (i.e. *depth* of propagation) of each tweet.
In this case, a directional path network consists of a network of users, N, of size n, in which each user N_i is followed by user N_{i+1} for $1 \leq i < n - 1$. As a result, all users in the network, except the user N_n, have precisely one follower.
Since each internal user only has one follower, the likelihood of a retweet occurring at each timestep is somewhat reduced, it is expected that the retweet volume will tail off more soon than in other network types. Propagation is also hindered by the fact that each retweet can reach an audience with a maximum size of 1 at each stage, thus relying on that single user making the retweet. Figure 4.1 shows the frequency distribution for a path network when simulated with the logistic regression over a series of tweets. The graph shows a very large proportion of single retweets, which reduces logarithmically with larger volumes.
The likelihood of a user getting the chance to retweet, and also deciding to retweet, becomes the product of a probability function the further the tweet travels, where user N_i requires all users N_0 to N_{i-1} to pass on the message before it even gets a chance to make the retweet decision.
As a result, if the retweet decision chance of each user is more or less equal, the chance of user N_2 retweeting the tweet is of an order of magnitude less than that of user N_1 deciding to retweet. The graph shows half life-style behaviour; owing to the fact that

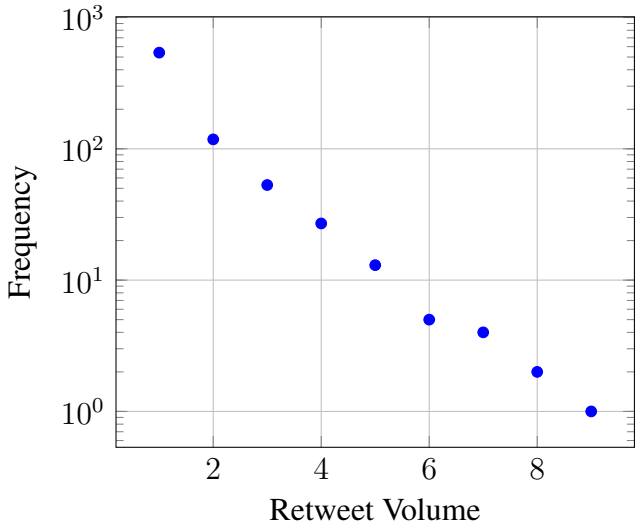

Figure 4.1: Retweet volume frequency distribution from path network simulation.

each retweet is exponentially less likely to occur than the previous retweet. Note that the graph is plotted on a log-linear scale.

4.4.2 Random Network

Random networks are more similar to Twitter's own social structure than path networks, but are a much more basic and uniform version and do not consider more influential users or the development of Twitter communities.

A random network is defined as a network of users, N, of size n in which a user N_x has probability p of following user N_y. Thus; as the probability p is increased, the likelihood of a user following other users in N increases, causing the overall network edge density to increase. Generally, the average number of followers and followees of a user is proportional to $p \times n$. Thus the parameters for constructing such a graph are the network size, n, and the attachment probability, p. The simulation results for the random network indicates a higher distribution of mid-range retweet volumes.

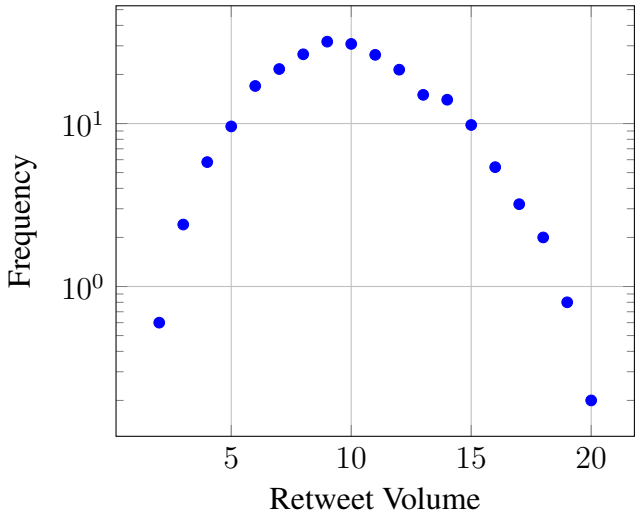

Figure 4.2: Retweet volume frequency distribution from random network simulation.

4.4.3 Scale-Free Network

Include more mathematical analysis of scale-free networks (in general) - i.e., in what way are they logarithmic?

A scale-free network is a network of users, N, of size n and is generated in such a way so that the resultant distribution of degree follows a power-law. *In*-degree signifies the number of inward edges to a node (i.e. the number of followers of a user), whereas *out*-degree is the number of outward edges (i.e. the number of users that user follows). Scale-free networks have been the subject of a fair amount of research, and are explained more thoroughly in [**?**]. In our implementation we use NetworkX[1], a Python networking package, to generate directed scale-free networks through a preferential-attachment algorithm based on the network size and edge density as parameters.

Figure 4.3 shows the frequency distribution of retweet volumes. Since the data is plotted on logarithmic scales, we see a logarithmic trend very similar to our results in [**?**].

[1] http://networkx.lanl.gov

4.4.4 Comparison to Real Twitter Data

In our previous work, [**?**], we captured and analysed data which contained results on the distribution of retweet group sizes. In that paper, a retweet group was defined to be a set of tweets containing one tweet and then all the retweets of that tweet. Thus the retweet volume looked at in this section is effectively the cardinality of the retweet group (minus one). Since the results in the above experiments also look at the frequency distribution of retweet volumes, then we should be able to draw some comparisons.

We compared the data produced by the different types of network to this previous data, and found that the scale-free network produced a distribution similar to that from the real Twitter data.

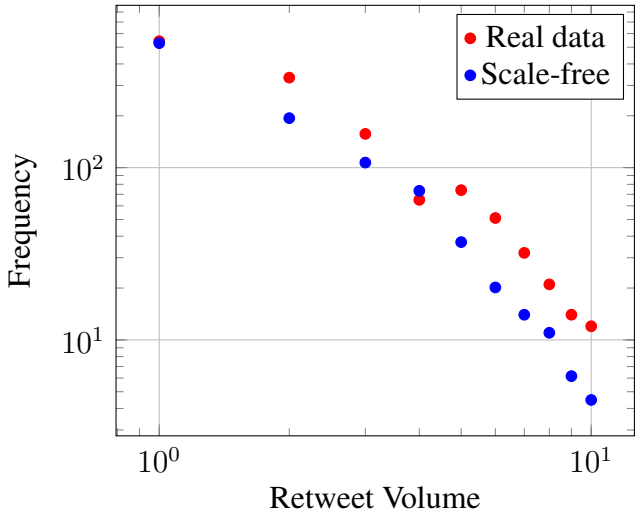

Figure 4.3: Comparing the retweet volumes distribution from scale-free graph simulation to data from Twitter's graph.

4.4.5 Structure Comparison

Each network structure has been demonstrated to show different propagation characteristics. This has shown that, in addition to a user's own retweet decision, the actual

spread of a tweet depends somewhat on how the author's local network is constructed. With lots of edges in the graph, there are many more paths down which propagation can occur, increasing the number of times a retweet decision is made, and therefore an increase in the overall number of retweets occurring. The retweet decision facilitated by the model, therefore, combined with a user network give an overall *retweetability* of a tweet that will vary depending on the network it's being propagated in, the source user, and the intermediary retweeters.

The path network, as designed to be the extreme case, has shown to allow poor propagation. Importantly, whilst the network parameters and retweet decision had to be globally increased to obtain any sensible data from this simulation, the trend still shows how propagation down a single chain isn't hugely effective.

The random network facilitated many more retweets due to the fact that users had a very similar in- and out-degree in all cases across the network. This means that each user is able to receive a lot of information, and is also able to pass on (whether an author or a retweeter) information to lots of users simultaneously in the graph.

Despite random networks supporting large retweet throughput (i.e. high *recall* of information), the disadvantage is that the interest *precision* is much lower. This is because this type of network relies on users following a large number of other users, thus meaning that they would receive more 'noise' (i.e. uninteresting tweets) than if they were more limited and selective. Although tweets that are retweeted are usually of a higher *quality*, not all retweeted tweets will be interesting to all users.

Finally, the scale-free network, whilst not having the highest throughput of information, does have trends most similar to the data on retweet distributions collected from Twitter's social graph. This is due to its ability to emulate more influential users and areas of dense communities (as discussed in [**?**]). These networks have the potential to allow for large numbers of retweets, especially if they are sourced from one of these more dense areas, but typically die off more quickly as the tweets are retweeted through less influential users.

4.5 Interesting Additional Findings

4.5.1 Graph Density

Define edge density (with equation

4.5.2 Results

Show links between follower audience -> local network -> density.

4.5.3 Uses

From this data, it is demonstrable that various parameters can be generally successfully inferred from very basic user information.

4.6 Predictions From the User Graph

The final part of this paper focuses on the ongoing development of a method to predict the interestingness of a tweet based on the work in the previous sections. The prediction method, at a high level, compares the predicted retweet outcome of a given tweet to the number of times that tweet has actually been retweeted. If, for example, a tweet is simulated with the help of the model and produces a prediction of two retweets, but the tweet has actually been etweeted four times, then we can infer that this tweet is more interesting (at least, to a subset of users).

The tweet and user features we looked at earlier in this paper are very static, binary features, which do not take into account the actual content of the text of the tweet. Therefore, if a tweet is retweeted more than was predicted, then there is something in the tweet, such as a link to a particularly interesting article or a breaking news story,

that makes it more interesting than the average tweet, with the same binary features, that was used to train the model.

In order to improve the fairness of the experiment, we wanted to ensure that the environment of the tweets (i.e. the user network they are propagated through) is the same as its real-life Twitter counterpart. We could then choose a user, which would become the source user, U_s, in the set U, and simulate that user's own tweets within their particular local network as described by the model above. This would then produce a retweet volume for this user's tweets, to which we could compare the number of times that tweet has *really* been retweeted.

4.6.1 Data Collection

Due to the exponential scaling properties of Twitter's social graph, it was infeasible to collect any more than two hops away from each user as a representation of that user's local network under the rate limitations of Twitter's REST API.

In particular, a single Twitter account running an instance of an application was allowed, at the time of these experiments, a maximum of 350 REST API calls per hour. One call would be required, for example, to obtain up to 5000 of the followers of a particular user (i.e. one follower hop from the user). An additional call would then be required to collect each of that user's follower's followers (in order to obtain the *second* hop from the source user).

Thus, a user who has 700 followers would require 700 API calls to collect that follower network, in addition to the one required to collect that source user in the first place, and would therefore take over two hours of collection. To collect the *third* hop from the source user would drastically multiply the number of required requests (even if there is significant overlap between the followers) and the time needed.

If each of the 700 followers of the source user had, on average, 200 followers, then this would require the gathering of 700 x $200 = 140,000$ users, equating to more than 402 hours of data collection. Bearing in mind that this would only collect the network

features for *one* user, it is clear to see how this is an impractical approach.

Luckily, in [**?**], we found that the vast majority of retweets occur *within* two hops of the source user (i.e. a path length of less than three), so we considered that the distance from the source user in each case would be sufficient.
In June 2012, the Twitter REST API was used to conduct a random walk through the social graph. For each user, we collected the most recent 300 tweets (including each tweet's metadata - particularly their retweet count) and their local follower network within two hops. We didn't collect the friend network, as we were only interested in tweets propagating outwards from the source user.
After processing that user, the walker chose a user at random from the present user's set of followers and made this the new current user from which to collect data for. If the present user, at any stage, does not have any followers, a list of previously accepted users is maintained and a follower is chosen from one of those instead.
The walker continued until the rate limit was met, at which time the current state was written to disk, and the walker waited until the rate limit was reset before continuing.
Generally, this resulted in, for each user, a set of up to 300 tweets (totalling to around 10,000 tweets in total) and the network in which these tweets were propagated within. There was no need to collect any further data to train the regression, since we were able to re-use the trained model we used earlier.

4.6.2 Validating Results

Needed to validate results using human input. Machines themselves are generally unable to express human interests, so results need to be properly evaluated.

Crowdsourcing

Discuss crowdsourcing, its uses, how it is useful in this area. Talk about its history (with any references), and then about mechanical turk.
Mention mechanics of mechanical turk, how it is US only (but we used crowdflower - which autmatically handles submission to MT and several other crowd-sourcing services.

What We Wanted to Assess

Used MK, etc.

Constructing the Questions

Set up questions (i.e. 5 tweets - choose most interesting and least interesting), give example of this.

In order to validate our prediction results, we ran a pilot user study in order to obtain some human input on the interestingness of each tweet. We compiled the tweet data into a set of questions which were submitted to Amazon's Mechanical Turk. Each question consisted of five tweets from our dataset and each Mechanical Turk Worker (MTW) undertook five questions. Each question asked the MTWs to select which tweet was the most interesting of the five, and which was the least interesting.
For consistency we ensured that at least three MTWs had answered each question. When selecting tweets to include in the Mechanical Turk questions, we excluded those which are '@-replies' - i.e. tweets which begin with another user's screen-name and typically form part of a conversation between two or more users. This meant that there were around 4,500 tweets in total in the questions.

Through using the model and simulating each user's tweets through their individual local networks we achieved around 86% accuracy in correctly predicting the number

of times each tweet was retweeted.

The precision in predicting the *interestingness* of each tweet was around 30%. While this value is low, it does mean that in 30% of cases, a tweet that we predicted to be interesting was verified to be interesting by at least three MTWs all selecting one tweet from a set of five. In addition, when simulating the questions by randomly choosing the most 'interesting' tweet of the five in each case, the performance was unable to near our precision even after several thousand iterations.

4.6.3 Improving This

Need offline methodology.

One route for this would be to try and infer a user's local network from a set of their immediate parameters, drawing on our earlier work suggesting that the Twitter network has the properties of a scale-free small-world graph. Through studying graph patterns, it is possible to make sensible inferences on the edges and nodes of a user's local network based on their follower count. From this, a graph edge density can be calculated, $d = \frac{|E|}{|N|(|N|-1)}$, for use in generating a scale-free network.

Since, for these preliminary experiments, we were only able to collect data from users with a more modest local network, the real and predicted retweet values were both relatively low, allowing more room for error. When simulating much larger local networks involving many more real retweets for each tweet, predicting interestingness, with some threshold value, may become more accurate and thus help improve the precision. The reason for this is that the retweet count of tweets that naturally get retweeted many tens, hundreds, or more times is likely to vary more with interestingness than those that are naturally only retweeted very few times.

4.7 Future Work

There is much further research that could be carried out based on the results in this chapter. Now that the foundation has been laid for simple retweet prediction based on network analysis, research could begin to look at ways in which, as mentioned, networks could be generated based on a few environmental features surrounding users.
This would allow for quick generation of user networks (bypassing the need for data collection) and would also support the same calculations for more highly influential users (users with more followers and more retweets per Tweet).

For this research, the notion of the network will continue and form the basis of the environmental features in the next chapter. Since we now know that the network plays an important role in dictating the way in which information can propagate

4.8 Summary

In this chapter we aimed to carry out a study on the behaviour of propagation through different types of social graph structures and to introduce our ongoing work into predicting the interestingness of tweets from their retweet patterns.
Using a set of tweet and user features, we trained a regression model which we used to simulate a number of tweets through different network types. We produced a distribution of retweet volumes for each network type and confirmed that, with the same tweet features, different network configurations do indeed facilitate different retweet behaviours in terms of propagation spread. We were also able to compare our results to data from Twitter to verify that Twitter's own social graph most closely resembles a scale-free small world graph.
We then finished by discussing how we used the trained model to simulate real networks from Twitter, along with the tweets that were passed through these networks, in order to try to predict how interesting a tweet is based on its retweet patterns. While

we were able to often correctly predict the retweet outcome of a tweet, we found that more work would be required to improve the performance of predicting whether or not these tweets are truly interesting to users.

Chapter 5

Inferring Interestingness of Tweets based on Information Flow Through the Network

Mention:

- How this chapter builds upon network stuff in previous chapter
- We hope to compare and contrast two better ways of predicting retweet volume *and* interestness
- What needs to be improved (speed, usability - more users with more followers etc.)
- Why do improvements need to be made?
- How is this useful, and how does first chapter relate to work done here?

Discuss:

- Does not use network to simulate tweets - instead uses a set of user features
- Previous chapter shown how basic features can be used to generate a scale-free network, which is what twitter is

- Use these features as input attributes of a new machine learning technique model.
- This method does not use a network or model individual user decisions
- Trained on a set of that particular user's tweets with the retweet outcome of integer type
- A new tweet modelled with the regression outputs a retweet volume prediction without having to simulate the Tweet's travels through the network.
- Discuss about the machine learning approach used (logistic regression and how it works)
- Talk about the 'binning' of retweet outcome volumes and its approaches (distribution dependent / independent, tables of precisions, etc.)
- Link 'retweet volume' to 'retweet group size'

Mis-calculation errors when validating the interestingness predictions: - caused by non-authorisation to collect the data from Twitter (i.e. user has a protected account). Therefore we cannot build the test (or train data) successfully for these tweets - if an experimenter selected one of these Tweets, then we have to discard that timeline from our analyses.

In this chapter, we use the features surrounding both the social structure of Twitter and the Tweets that propagate within it to develop a methodology for deciding upon *which* Tweets may be of specific interest, and also an introduction to inferring *how* interesting the information is. The work does not take into account the relevance of a piece of information to a certain user, but instead the general interestingness level of the Tweet. The work focuses on the difference between the raw popularity of a Tweet, demonstrated by its retweet activity, and how interesting the Tweet actually is to its recipients. While it has been shown that training a model with many Tweet features can be accurate in predicting how many times a Tweet might be retweeted [**?**], here we focus on the notion of Tweet content beyond those static features. That is to say, when comparing

the popularity of Tweets, that there is some content (either in the Tweet itself or in a webpage or image at a URL included in the Tweet) that makes that particular Tweet stand out and to cause *affective stimulation* [**?**] to viewers.

The focus of this work is in quantifying the interestingness of Tweets; that is, the universal relevance level of the information contained (or portrayed through links or media) in a Tweet. We infer this interest level by analysing the *retweet volume* of Tweets in addition to their surrounding static features and the features of its source user and the network it propagates within. The retweet volume of a Tweet is defined as the total number of times that particular Tweet has been retweeted.

This paper considers only Tweets that have been retweeted using the 'button' method, which is a single-click function on the Twitter website and its applications to carry out a retweet. Twitter users also sometimes imitate a retweet by copying the text of the original Tweet and prepending it with an "RT" followed by the original author's screen-name, which allows them to add annotations to the Tweet if they desire.

In this work, we introduce a method of calculating if a Tweet is interesting, or not. This is done on the basis of a comparison between a predicted retweet outcome for a particular Tweet and the *observed* retweet volume of the Tweet. The method relies on the fact that the content of a Tweet is an important factor in retweet decisions, but avoids having to do much processing of the Tweet's text contents themselves (such as through natural language processing or sentiment analysis).

To determine *how* interesting a particular Tweet is, we introduce an 'interestingness score', which is based on the quantitative *distance* between the real and predicted retweet volumes for a specific Tweet. The method and scoring is discussed in more detail later. If the interestingness score for Tweets can be determined and is known for a set of Tweets, then this method could be used as the basis for an information retrieval or delivery system, where relevant and interesting information can be shown to users without them having to know about it (e.g. follow the right set users) or search for it first.

5.0.1 User Influence

Of importance to this work is the difference between the retweet volume and the interestingness of a Tweet, and the fact that one does not necessarily indicate the other. The most obvious case in which this occurs is where there are influence discrepancies between users. For example, one of the most influential Twitter celebrities is Justin Bieber (@justinbieber), who, at the time of writing, has nearly 40 million worldwide followers and achieves an average of around 50-120 thousand retweets per Tweet. His Tweets currently rarely receieve less than 40,000 retweets.

Average Twitter users generally attract a couple of hundred followers and would normally receive very few (if any) retweets per Tweet. A particularly interesting Tweet from such a user may be retweeted around 5-20 times (though this depends entirely on the level of influence of a user and their number of followers).

It is therefore apparent that an uninteresting Tweet from Justin Bieber could get retweeted 50,000 times and an exceptionally interesting Tweet from a less-influential user may get 30 retweets, and that the retweet volume, in this case, really isn't indicative of the *interestingness* of the particular Tweet.

Of course, this brings about the notion of information relevance, and the fact that a Tweet could be very boring or irrelevant to one user, and very interesting to another. In this work we focus on *global* (or 'average') interest, where retweet scores and predictions are made for the general case. It is our belief that Tweets that are retweeted more than expected within their authors' local networks (relative to the authors' own regular output) are also likely to be of interest to a wider audience.

5.1 Interestingness Scores

As discussed, our proposed process for deciding upon a Tweet's interest level involves conducting comparisons between the number of times a Tweet has been retweeted and the *predicted* retweet volume of the Tweet. The premise behind this is that there

is something beyond the static features of a Tweet (e.g. whether, or not, the Tweet contains a URL, the length of its content, the influence of its author, etc.) that causes a specific Tweet to be more interesting than another with precisely the same feature set. The retweet count of a specific Tweet is returned, at the time of writing, as part of a standard call to v1.1 of the Twitter REST API, and includes the total number of times that the original Tweet was retweeted using the button retweet method.
Essentially, our method follows these basic steps:

1. Obtain and extract contextual information on a given Tweet, t,
2. Generate predictions on the retweet volume for t,
3. Compare the predicted values to the *observed* retweet count of t,
4. Make inferences based on this difference between the predicted and observed values.

We use Bayesian Network classifiers, trained on a set of Tweet and environmental features, in order to make predictions on a Tweet's retweet volume. In this work we consider two of these classifiers for making predictions for each Tweet; one trained on the Tweet's source user's own Tweets and another trained on a 'global' corpus of Tweets (a larger dataset of Tweets from many source users). Thus, we obtain two retweet volume predictions for each Tweet.
For a Tweet, t, we define the observed (or 'real') retweet outcome to be $T_O(t)$, the predicted outcome using the global corpus model to be $T_{P_G}(t)$, and the predicted outcome using t's author's user-centric corpus model to be $T_{P_U}(t)$. The scores for t's actual outcome compared to the predicted outcome using the global and user-centric models, $TS_G(t)$ and $TS_U(t)$ respectively is calculated thus;

$$TS_G(t) = \frac{T_{P_G}(t)}{T_O(t)} \quad TS_U(t) = \frac{T_{P_U}(t)}{T_O(t)}$$

This provides a positive score where;

$$TS_G(t), TS_U(t) \begin{cases} > 1 & \text{indicates } t \text{ is interesting} \\ \leq 1 & \text{indicates } t \text{ is non-interesting} \end{cases}$$

$|TS_G(t) - 1|$ represents *how* interesting (or non-interesting) t is and $TS_{Avg}(t)$ denotes the mean of its user and global score.

This method relies on collecting retweet data from Twitter and involves taking a snapshot of Tweets at one stage during their lifetime. Since Tweets do not decay (unless they are deleted by their author), they can be found and retweeted by users at any time after their composition. In this work, we assume that the significant portion of retweet activity has already occurred for the Tweets that have been collected. Indeed, [**?**] carried out temporal analyses on retweet behaviour and discovered that, on average, 75% of the retweets of a particular Tweet occur within one day and that 50% of retweets take place within one hour of the Tweet being posted.

Based on this, to ensure that the retweet count mostly reflected the 'final' retweet count, only Tweets more than one day old were considered for experimentation.

5.1.1 'Twitter is a Memepool'

The term 'meme' was defined to be a "unit of cultural transmission" [**?**] and the notion of memetics is an analogy to biological genetics. Unlike genes, memes are entirely non-physical and represent a cultural idea or other human-based behaviour.

Genes survive and are passed on through *replication*, and this replication occurs more frequently and efficiently when they are more suited to their environment. A gene's genome is the set of information that represents, in its entirety, the features that make up that particular gene (such as eye colour, height, some aspects of personality, and so on).

Memes are similar in that they contain a set of features, such as the wordings of a phrase or their relevance to other cultural aspects, which cause them to be less or more likely to be replicated in different environments. For example, an Internet meme relating

to the Star Wars movies would likely have a greater chance of reproduction (through reposting, discussion, etc.) in an environment of sci-fi fanatics than when amongst more mixed groups.

The meme is a useful analogy for the work in this paper, since it also helps outline the way in which Tweets are replicated within Twitter. Like a meme, a Tweet has a specific set of features (the text it contains, any hashtags, the inclusion of mentions, and so on) and it exists within an environment consisting of a set of interconnected users on the Twitter social graph. A particular Tweet would generally have a greater chance of being reproduced, through retweeting, in certain networks than others. Tweet features are similar to the *genome* of a gene and the features of the network the Tweet exists in (i.e. the users that receive and have the opportunity to assist in a Tweet's propagation) form the *environment*.

More:

- Gene is a physical entity containing information and instructions. It is a unit of genetic inheritance (i.e. offspring typically have a mashup of the genes of the parens)
- The result of the data held by a gene (the genome) means that organisms with certain genes are able to reproduce and survive more than other organisms containing different genes.
- Thus the gene is able to replicate under certain gene- and environmental-centric conditions.
- A meme is similar to gene but is non-physical. They are a unit of cultural inheritance (an idea, phrase, behaviour, etc.).
- Like genes, memes are able to survive better when their features (*menome*) are suited to the meme's environment. In such environments, the meme is able to be shared and replicated more efficiently and frequently.

- A Tweet, again, is similar to both. A Tweet itself has many features (the text of the Tweet, the time of its origin, its length, etc.) and their environment, the Twitter social structure, has features (namely the users that belong to it and the way they are connected) which may facilitate the replication (i.e. Retweet) of the Tweet.
- A Tweet existing in different social structures will have different Retweet patterns, which is what we want to show in this chapter.
- Thus tweetfeatures = genome, userfeatures=environment

5.2 Retweet Volumes as Nominal Attributes

In order to try and improve the accuracy of the model at predicting retweet volume outputs, a nominal output attribute would be better than a real one. Predicting a continuous numeric value could render inaccurate results and calculating the cut-off points at which to mark as the upper- and lower-bounds for the output based on the inputs would raise difficulties.
Instead, the retweet outcomes were to be 'binned' into several categories which would be determined on the fly based on the outcomes present in the training data.

In the global model and each of the user models, the retweet volume was trained as the *outcome* feature. Accurately predicting continuous values is a challenge for many machine learning algorithms due to the way the model is built around the features. To assist with this in our experimentation, the retweet outcomes were 'binned' into nominal outcome categories.
The binning algorithm needed to be dynamically based on the distribution of retweet outcomes so that each bin was of a roughly equal size.

5.2.1 Binning the Retweet Outcomes

Describe the different methods (linear, distributed 1, distributed 2), and their advantages/disadvantages, with examples showing the graph of what the bins look like.
Focus on the distributed 2 example, and why this is better. 'Requested' bin number not usually the same number as what is actually returned (due to large numbers of smaller retweet groups).
Pseudocode

In [**?**] it was shown how the general distribution of retweet volumes forms a long tail with a very high proportion of single retweets and for which the frequency drops off considerably for larger retweet volumes. As a result, linearly binning the outcomes would reduce the number of feature instances for higher retweet outcomes and thus cause the Bayesian Network classifiers to be considerably less accurate.
The binning algorithm that was used involved calculating the projected size of each bin from on the total number of feature instances and the desired target number of bins. The bins were then filled accordingly, such that each retweet volume frequency would only appear in one bin. For example, The feature instances with zero retweets would all appear in the first bin, no matter how many there were. Instances with larger retweet volumes would then typically share a bin with instances with similar retweet volumes. Each bin 'range' (i.e. the range of retweet volumes of instances contained in the bin) represents a nominal category of the retweet volume outcome in the Bayesian Network, and each feature set instance is associated with precisely one outcome category.
Since the binning algorithm is dynamic, the bin sizes and ranges vary from dataset to dataset. As a result, the categories in each user-based dataset are different from one another, reflecting the different number of retweets that each author's Tweets are likely to receive.

5.2.2 Varying Bin Sizes

Number of bins: explain how accuracy worsens as bin number increases.

5.3 Techniques Used

For the work presented in this chapter, various techniques and approaches were used for handling the data and for generating the scores. The most notable are now assessed in this section.

5.3.1 Machine Learning

Explain about Machine Learning, its uses, techniques and how this is useful.
Talk about how it was used in previous chapter, but that more in depth here. Bayesian Network

Classification Performance

Compared several types of classifiers (show table comparing accuracy, etc. of different types)
Explain that Bayesian Network is best (quick, accurate)

Training Results

Playing with Weka to improve the prediction performance (i.e. different number of bins, different features)

5.3.2 Crowdsourcing

Crowdsourcing is a technique often employed by companies and researchers to collect large amounts of data from a (often large) distribution of people or devices. Examples of this in practice include the use of surveys and user-contributed reviews (such as in Google Maps).
In this paper we use Amazon's Mechanical Turk service as a method for crowdsourcing interestingness validations from many people, who are known as Mechanical Turk Workers (MTWs)

5.4 Experimentation

In this section we describe the stages involved in making interestingness inferences of Tweets based on their individual retweet behaviour. We start by discussing how the data was collected and how the information was then processed. We go on to illustrate how we employed our Bayesian Networks and the features we extracted for training them in order to make the retweet predictions. We finish by discussing how we then infer the Tweet interestingness and how this was validated.

5.4.1 Data Collection

In March 2013, a random walk was carried out through the Twitter social graph using Twitter's REST API, originating with one of this paper's author's Twitter account. Each step of the walk involved focusing on one Twitter user, collecting information on that user, and then selecting a random follower of that user. This follower then became the focus of the next step.
At each step, a set of the most recent Tweets from the current user were collected. The number of Tweets returned by the Twitter API differed from user to user, based on their recent Tweet-posting frequency, though usually a few hundred Tweets were

yielded. In addition to their Tweets, we also collected information on the user itself and on a sample subset (up to 100, if they exist) of its friends and followers. We used a sample instead of collecting information on *all* of a user's friends and followers so that we could maximise the efficiency of our method in terms of data collection. It also meant we had a snapshot of an additional 200 users in the author's local network to give the classifier a notion of the activity of this local network both upstream and downstream from the author.

This gave a dataset containing around 241,000 Tweets from 370 unique Twitter users and, of those Tweets, around 90,000 were cases where the retweet volume was greater than zero. The dataset was split into two datasets: a training set (90%) and the testing set (10%). The original dataset was split in such a way as to allow all Tweets belonging to one particular user to exist in only one of the two smaller datasets. The training set was used to train a model, as described below, and was then discarded from the rest of the experimentation.

5.4.2 Data Corpora

As discussed in the methodology section, we are interested in producing *two* predictions for each Tweet; one when compared to the global set, and another from comparisons to the rest of that user's Tweets. Therefore, a new dataset was formed for each user in the testing dataset which contained the Tweets only from that particular user. For the remainder of this section, the testing dataset is the *global* corpus of Tweets, and each of the individual user datasets are known as *user* corpora.

The Bayesian Network was chosen as our classifying algorithm to produce the predictions as it was suitable for the data types of the Tweet and user features and was also found to perform efficiently in terms of precision and recall across the outcome predictions when carrying out test cross-validations against itself. The prediction precision weighted across the outcome categories was calculated to be around 70% in cross-validation tests on the training dataset features.

5.4.3 Features

To train the Bayesian Network model, a series of new features were harvested. Generally, each of these features fell into one of two categories; user features and tweet features.
The Tweet features follow the same ideas as the features used in the previous chapter: static, generally binary features that describe the structure of the Tweet. The user (or 'network') features are related more to the *network* to which the Tweet belongs.

Features were extracted from the Tweet and user data to train the Bayesian Networks. Each Tweet is represented by an *instance* of feature sets specific to that Tweet (and its author, if appropriate). In each instance the outcome feature is the retweet volume, which is categorised using the technique discussed later.
Since some features are static among Tweets from the same user (for example, a user's follower count), a different feature set is used to train the user models and the global model, as described below.

Features for the global corpus model

A total of 31 features were used to train the model classifier from the global data corpus.

The features used to train the global model are outlined in Table 5.1. The network features apply to both the followers and friends retrieved for each author. For example, the first feature of this category, 'max. follower count', represents two features referring to the maximum follower count observed in the sample of the author's followers and in the sample of the author's friends.
While Tweet features are permanent after the Tweet has been created, the author and network features are dynamic in that the social graph is of a constantly altering structure as links are formed and broken between users. However, in this work, we assume

Feature category	Feature	Feature data type
Tweet ('genome')	mention	{True, False}
	Tweet length	real (numeric)
	url	{True, False}
	hashtag	{True, False}
	positive emoticon	{True, False}
	negative emoticon	{True, False}
	exclamation mark	{True, False}
	question mark	{True, False}
	starts with 'RT'	{True, False}
	is an @-reply	{True, False}
Author	follower count	real (numeric)
	friend count	real (numeric)
	verified account	{True, False}
	status count	real (numeric)
	listed count	real (numeric)
Network ('environment')	max. follower count	real (numeric)
	min. follower count	real (numeric)
	avg. follower count	real (numeric)
	max. friend count	real (numeric)
	min. friend count	real (numeric)
	avg. friend count	real (numeric)
	avg. status count	real (numeric)
	proportion verified	real (numeric)

Table 5.1: Features used to train the model from the global data corpus

that the changes to these features are not significant over the Tweets collected for each user and try to minimise this by only considering recent Tweets of each user.

Features for individual user models

The 10 Tweet features were used for training each user-based Bayesian Network to reduce redundancy in the model since the author and network features would be the

same across all of the Tweets from one user.

5.4.4 Making predictions and inferring Tweet interestingness

The global and user models were then trained using the features to produce one global model and one user model for each unique author in the testing dataset, as described above.

To produce the global predictions, T_{P_G}, each Tweet in the testing database was evaluated against the global model, which output a predicted value for the bin that Tweet is predicted to belong to.

The user predictions, T_{P_U} were calculated in the same way: each Tweet in the testing dataset was evaluated against the model based on the Tweet's author.

Each Tweet then each had two retweet volume predictions. The upper bound of the binned category assigned to each prediction was compared to the *observed* retweet count to produce the two interestingness scores, TS_G and TS_U, as discussed earlier.

5.4.5 Planning the Result Validation

To assist in validating the interestingness scores produced for the Tweets, tests were required in which humans would be responsible for also assessing how interesting they feel the Tweets were. The predicted interesting Tweets could then be compared to the Tweets chosen as interesting through human judgement.

For this stage of the experimentation, certain Tweets (and users) were removed from the dataset to be tested by the humans. Since the Tweet data was collected using a random walk through Twitter, there was no governance over the content of the text in this data. Therefore, users who frequently used offensive or non-English (since the Tweets were to be validated by English-speaking individuals) language had their Tweets removed from the test set. In addition, individual '@-replies' (Tweets starting with the '@' symbol) were also stripped from the test set since these usually denote

conversations on Twitter between two or more users and are unlikely to convey any interest to outsiders.

Validating the Results

In this context, MTWs have no connection to the Tweets in the dataset, and are random users of Amazon's Mechanical Turk who decide to take the job on. Collecting the validation data in this way, and instead of using a small set of people, means that we can collect a diverse opinion on Tweet interestingness from people whom the Tweets will have varying relevance to. This helps reinforce the notion of *global* interest levels, where Tweets can be interesting to a large number of users.

The human validations were carried out such that the MTWs were presented with questions, each of which consisting of five different Tweets from one specific author. Each question asked that the MTW select the Tweets they consider to be the most interesting of the group and that they must select at least one. MTWs were paid $0.05 for each question they answered and there were 91 unique MTWs in total who responded to the questions.

The tests were under the conditions of a randomised controlled trial, such that each Tweet was assessed in three different contexts (i.e. each Tweet would appear in three different questions alongside four randomly chosen other Tweets) and that each question would be responded to by three different MTWs.

To represent the validation for this ongoing work, 750 Tweets were selected, at random, from the stripped testing dataset and were organised by user to ensure that Tweets only appeared in questions alongside other Tweets from the same author. Since each Tweet was required to appear in three different questions and each question consisted of 5 unique Tweets, this resulted in a total of 450 unique questions, each of which was answered by three different MTWs.

5.4.6 Validation Results

From the 450 questions asked of the MTWs, 325 questions had responses where the response was chosen with a confidence of two-thirds or greater. Since MTWs had the opportunity to choose more than one Tweet to be the most interesting, 349 Tweets (the subset of all tested Tweets), denoted as T, were selected as sufficiently interesting. Tweets selected from individual questions by only one of the assessing MTWs for that question were discarded.

General Performance

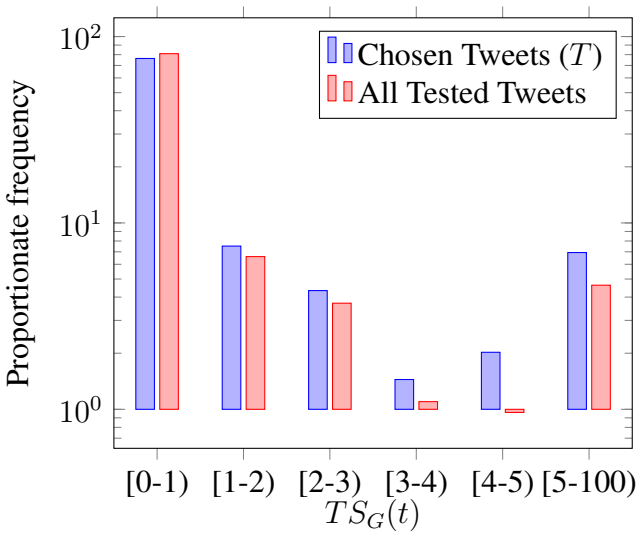

Figure 5.1: Proportionate frequency distribution of global scores in the entire test set compared to only those in the subset T..

In the subset T, our method predicted 140 Tweets to have $TS_G(t) > 1$ and 80 Tweets where $TS_U(t) > 1$. These results were validated by the MTWs in that they agreed that all Tweets $t \in T$ where $TS_{Avg}(t) > 1$ were interesting in 60% of cases, with the performance of the global scores, $TS_G(t)$, achieving 65% precision. The user scores, $TS_U(t)$, were less accurate and yielded a precision of around 55%.

In addition, we show how the proportionate frequency of Tweets with higher values of

$TS_G(t)$ is greater in the subset T than in the entire set tested with the MTWs (Figure 5.1). This also demonstrates that Tweets with low global scores ($0 \leq TS_G(t) < 1$) are more frequent in the entire test set than in the subset T.

Per-Question Performance

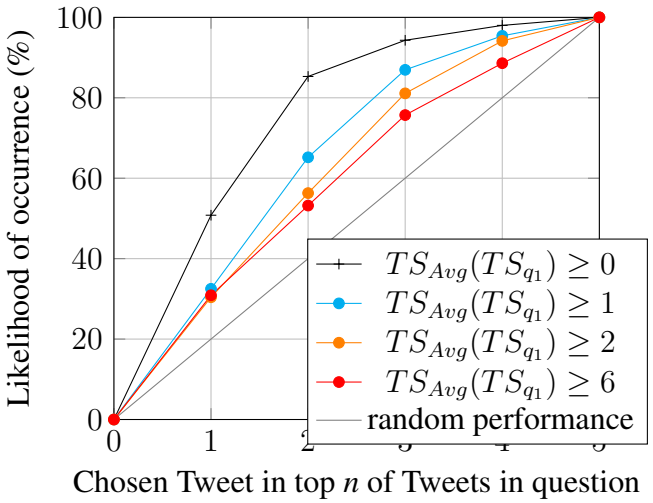

Figure 5.2: The probability of the MTW's chosen Tweet's $TS_{Avg}(t)$ being in the top *n* of Tweets for that question while varying the minimum allowed maximum Tweet score of the question..

Analyses were also conducted on the performance of the predictions on a per-question basis. Hereafter, a question, $q \in Q$, is defined as a set of Tweets where;

$$q = t_1^q, t_2^q, t_3^q, t_4^q, t_5^q$$

and

$$|q| = 5$$

For conducting these question-based analyses, each question's five Tweets were ranked in order of ascending mean interestingness score such that $TS_{Avg}(t_1^q) \geq TS_{Avg}(t_2^q) \geq ... \geq TS_{Avg}(t_5^q)$. We then calculated the number of times the MTWs chose a Tweet that appeared in the top n of the ranked list of Tweets, as shown in Figure 5.2.

In this figure, we vary the minimum allowed value of $TS_{Avg}(t_1^q)$ (the highest Tweet score in question q) to show how detecting more interesting Tweets was more accurate when the range of scores in each question is more disparate. We show, for cases in which $TS_{Avg}(q_1) \geq 1$, that the likelihood of the MTWs choosing one of our two most highly ranked Tweets of the question using the interestingness predictions is around 66% and the chance that they choose one of the top three ranked Tweets is 87%.

Probability of Selection

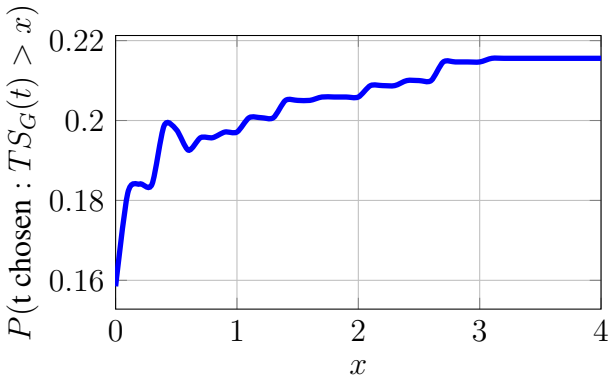

Figure 5.3: The probability that Tweet t is chosen given that $TS_G(t)$ is greater than a given value, x..

Finally, we show that the probability of MTWs deciding a Tweet t is interesting becomes higher as the value of $TS_G(t)$ increases.

In Figure 5.3, although we exclude cases of Tweets with $TS_G(t) > 4$ to reduce noise (due to fewer samples), a significant increase in probability is observed, particularly in the interval 0-1 representing the range of Tweets that are uninteresting (fewer observed retweets than predicted) to 'as expected' (observed retweet volume is equal to predicted). It is also illustrated that, from this initial work, Tweets with a predicted interestingness score of 3 or more are not significantly different from one another in terms of their 'real', human-judged, interestingness. However, more work will be carried out towards this in the future so that more accurate research can be done on Tweets

with scores of greater then 3 in this context.

5.5 Further Analyses

The interestingness scores have been validated in terms of there being recognition that they can signify interesting Tweets. This section will now continue onto some deeper analyses of the results in order to show *how* it is able to work.

5.5.1 Discerning Interesting Information from Noise

In this subsection, the human-selected interestingness selection will be assessed. Of particular interest is the likelihood of humans agreeing on an interesting piece of information and the properties of the Tweet scores in questions when agreements *are* made.

For this, analyses were made into the *disparity* of scores for Tweets. That is, the range of scores of Tweets in a particular question and the effect this has on human decision in deducing interesting information.

Num. confident answers in q	min. $d^{Avg}(q)$	max. $d^{Avg}(q)$	avg. $d^{Avg}(q)$
0	0	846	**17.6**
> 0	0	1445	32.1
1	0	1445	**34.3**
> 1	0	4	0.647
> 2	0	0.55	0.204

Table 5.2: Absolute $TS_{Avg}(t)$ disparity of questions with varying number of confident answers made. Entries in bold are used to highlight interesting values..

The absolute Tweet score disparity for a question, q, is defined as $d(q)$. For example, for a disparity of average scores, this is calculated thus;

$$d^{Avg}(q) = \max(TS_{Avg}(q)) - \min(TS_{Avg}(q))$$

Table 5.2 shows how the values for $TS(q)$ vary for questions with differing numbers of confident answers. A confident answer, as mentioned previously, is one where at least two MTWs have agreed on an interesting Tweet.

The data shows that the average $TS(q)$ is around double in cases where a question is answered with precisely one confident answers than in cases where there are no confident answers made. This indicates that a wider scale of interestingness in a question is useful for humans for picking out the content they'd prefer to read. If several pieces of content are more similarly interesting (or, as the case may be, uninteresting), then it becomes more difficult for an agreement to be made on which information is the *most* interesting.

In addition, the average score disparities in cases where multiple confident answers were selected are very low. This helps to reinforce the notion that pieces of information that are very similar in terms of interest level make it hard for users to decide on the *most* interesting. Indeed, in questions where this is the case, MTWs have selected, and agreed on, multiple Tweets.

To take this further, it is demonstrable that the probability of a confident selection being made for a particular question, q, increases as $TS(q)$ also increases (Figure 5.4. Thus, this reinforces the notion that people find it easier to discern interesting information when compared to *un*-interesting information. This, too, is highlighted in Table 5.3, in which it is shown that amongst *all* questions (i.e. not only questions that have been confidently-answered) the score disparity is much smaller between Tweets that were selected than the score disparity for the entire question.

This is particularly the case in which there are a few Tweets which have similarly high scores amongst Tweets which collectively have *lower* scores. Therefore, selecting confidently from the few Tweets with the similar scores become difficult, but it is demonstrated that these at least are *more* interesting than the ones that weren't selected.

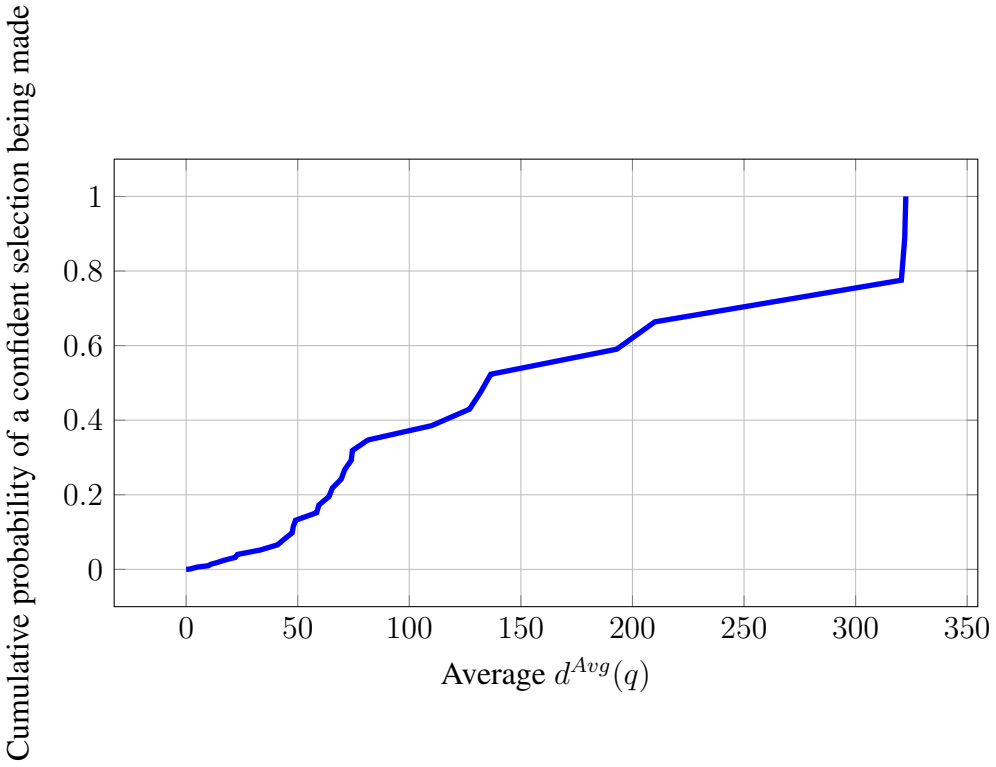

Figure 5.4: The probability of a confident selection being made for question q with varying $d(q)$..

	$TS_G(t)$	$TS_U(t)$	$TS_{Avg}(t)$
$TS_d(q)$	62.4	4.7	33.3
$TS_{d_C}(q)$	35.3	3.1	19.0
Ratio	57%	66%	58%

Table 5.3: Score disparity comparison between selected Tweets of question q and *all* Tweets in q when using the three different Tweet scores as metrics.

For example, this data shows that, on average, the global score disparity for selected Tweets of a question, q, was only around 57% that of the entire disparity of q.

Chapter 6

Critical Assessment and Conclusions

6.1 Critical Analysis of Results

6.1.1 Analysis of Initial Research

- What was the use of initial research?
- Are the results sensible?
- How have the results shaped the further research?

6.1.2 Analysis of Final Results

- Have methods been able to sensibly predict retweet volumes?
- Have methods sensibly inferred Tweet interestingness?
- What might have worked better?
- Which parts were useless?
- Which parts helped develop other areas of research which may have provided further avenues of research ideas?

6.2 Further and Future Work

How can this research be taken further in the future?

- Use previous results to predict how far a tweet is likely to be retweeted (for advertising purposes)
- Useful for detecting the kind of messages that are likely to travel further
- As well as providing an interest level, the systems also predict sensible estimations on retweet volumes.
- Perhaps useful for measuring the spread of rumours.

6.3 Conclusions

6.3.1 Summary

Summarise events and processes covered, reiterate what the point of the work was and how each part of the work covered relates to that.

6.3.2 Contributions

Restate the original contributions (from Introduction section). Explain the ways in which the work done relates to the projected contributions, that it is novel and useful.

Appendices

Source code, further diagrams, ideas, etc.