ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

TIK Institut für Technische Informatik und Kommunikationsnetze

Semester's Thesis

Inferring Shared Interests from Tweets

Konstantinos Chaloulos

September 30, 2011

Supervisors:	Theus Hossman	hossmann@tik.ee.ethz.ch
	Dr. Franck Legendre	legendre@tik.ee.ethz.ch
	Prof. B. Plattner	plattner@tik.ee.ethz.ch

Abstract

Social networks have intruded into the every day's life. Users contribute with uploading content and sharing it with friends and public. From this information knowledge on who know whom, who communicates with whom, who meets whom and who shares the shame interests with whom can be extracted. These four aspects of human behavior can provide valuable insights while designing networking algorithms and protocols for opportunistic networks. Having efficient algorithms for exchanging information can help in the decentralization of the current centralized approached used to store the information. The distribution of the information could be quite useful in cases where the infrastructure is absent and thus fails to provide services. So far research on the first three topics and the relations between them has be done.
In this thesis an effort on the introduction of how sharing the same interests correlates with the first three topics is done. In order to be able to do such a correlation, a measure on assigning similarity scores between users with respect to their shared interests needs to be created. In this thesis the problem of text comparison is investigated. A frequency of words based approach is chosen for the comparison of the texts posted on the social network called "Twitter". Five metrics are created based on special characteristic of twitter texts. The validation of the metrics is done in a set of 250 users. The results show that it is feasible to assign similarity scores between twitter users and thus provide a tool for measurements with the ultimate goal to deepen the understanding of how a pairs of users sharing same interests correlates with knowing each other, meeting each other or communicating with each other.

Contents

List of Figures

Chapter 1

Introduction

Nowadays more and more people register for themselves on online social networks (e.g. Twitter, Facebook LinkedIn, etc.). These networks provide platforms to people to post online texts concerning their activities and their interests. Furthermore, some more involved social networks allow the users to check in at some places (e.g. Foursquare, Gowalla). In other words users can broadcast their location in order their friends to be able to see that and get informed about their activity. These posts are saved in databases leaving traces behind concerning some personal preferences.
This externalization of personal information is already exploited in a way so that the advertisements can be addressed to people who is more likely that are interested in those products. Towards the same direction, but with non profit concept, the current semester thesis aims in measuring the similarity with respect to shared interests between two users of the social network "Twitter".
The degree of similarity between two users could be realized by different means. For this research thesis a text comparison method is chosen, that is the similarity between two users is measured by the texts that they post online. A pure text comparison to extract a similarity score is a problem that has a long history in artificial intelligence, philosophy, psychology. Several techniques and methods have been suggested and investigated, but none has dominated as the one and only solution. Thus during this thesis a research on related work was made prior to the implementation of the chosen one.

1.1 Motivation

The motivation to write this thesis comes from the field of Opportunistic Networks.Opportunistic Networks [1], [2] use human mobility and consequent wireless contacts between mobile devices to disseminate data in a peer-to-peer manner (via Bluetooth or WiFi Ad Hoc). Such networks can keep information flowing in case of lack or outage of mobile communication infrastructure (3G, WiFi). These situations may include an event of natural disaster (flood, earthquake, etc.), political censorship (Egypt, Libya) or in rural regions where operating infrastructure networks is not profitable.

Designing appropriate networking algorithms and protocols (for example routing protocols) for Opportunistic Networks is challenging, as it requires understanding patterns of

1. mobility (who meets whom so forwarding opportunities)
2. social relations (who knows whom so trust, altruism to forward data)
3. communication (who communicates with whom so need to forward data)
4. interest (who is interested in the same content so need to forward data)

So far, research is limited to considering these dimensions separately. Only recently, few studies look at how the individual relations relate to each other [3], [4], [5], [6]. However, these studies consider at most three of the described relations [5], [6]. The goal of this project is to extend the

Figure 1.1: Introducing "Shared interest" in the triangle of human behavior

study described in [6] with the fourth dimension: interest. While mobility, social and communication ties can be observed more or less directly with the help of databases provided by some of the social networks (e.g. Twitter, Facebook, Gowalla), common interests can be inferred indirectly from the similarity of the content of two users' posted texts.
The goal of this thesis is to develop a metric of how similar two users are, given the history of their Tweets. This metric can be compared to similarity metrics from the other three dimensions (mobility, social and communication). Hence after the realization of figure similarity metric it would be more feasible to investigate the relationship between the fourth introduced cycle "interest" of the 1.1 with the other three dimensions of the human behavior. The research project's ultimate goal is to have further understanding on how two users that share the same interests increase the probability of:

- wanting to communicate with each other, that is the communication cycle,
- knowing each other, that is the social cycle,
- meeting each other, that is the mobility cycle.

This deeper understanding will provide some more assistance in the design of more efficient networking protocols and algorithms for opportunistic networks. Moreover in this semester thesis we only focus on interest similarity between Twitter users.

1.2 The Task

The tasks as provided by the plan of the thesis include a first familiarization with the topic, a design of similarity metric and its validation. The three tasks are analyzed in further details as follows:

- Familiarization: The student familiarizes with (i) the topic, from the provided literature references (ii), the dataset of 25.000 Twitter users and their Tweets (provided in a MySQL database) (iii), additional tools for measuring similarity of Tweets, and (iv), tools

for statistical analysis of the data (e.g., Matlab or R).

- Similarity metric: The student develops a metric, which assigns each pair of users a score of how similar the content of their Tweets is. Such a similarity metric could for example look at string similarity, at frequency of words, etc.

- Validation: In order to validate the developed metric, the similarity score should be compared to some reference similarity of a subset of the Twitter users. For example, this reference could be a subset of users for which their interests are known (from Twitter list membership, etc.).

1.3 Overview

The thesis report is organized as follows: chapter 2 gives the reader an overview of state-of-the-art solutions to compare the similarity of text-based documents and to extract a similarity score between them; chapter 3 specifies how the text comparison was realized for twitter texts and which metrics are defined; chapter 4 explains the validation procedure, the tools used and the results and chapter 5 presents the future extensions and work that need further research.

1.4 Contributions

In this thesis, we developed the following contributions:

1. Assessing the relevance of content-based versus word frequency approach for tweet similarity comparison
2. Design of twitter specific similarity metrics
3. Implementation of the metrics and extraction of similarity scores between twitter users
4. Analysis of the difficulties and peculiarities encountered to implement and evaluate the solution

Chapter 2

Related work on Text Comparison

In chapter 1 the need to extract a similarity score between two twitter users was explained. Apart from the social connections of the users, information can be exctracted merely from the texts posted by the users. Text comparison and similarity exctraction is already a well discussed and research field of information retrieval. Several approaches can be found in the existing literature and in online guides.

2.1 Content Based Approach

One well known approach for extracting similarity between words, sentences and texts is the content based approach. This approach includes examining the several meanings of the words appearing in a document, comparing and extracting at the end a final score between two documents. Towards this direction a tool that can used is Wordnet.

2.1.1 Wordnet

Wordnet [7] is a lexical database of English. A huge effort for collecting English terms and then examining the relations between words was done by the creators of the Wordnet project. Groups of words which have the same meaning are formed and they are call "synsets". Each synset expresses a specific concept and different synsets are examined for their inter-relation in a semantic level.
Wordnet could be thought as a combination of a thesaurus and a dictionary. Applications using wordnet allow for document classification, searching and querying and other language processing problems, such as comparison. The ultimate goal of Wordnet is to somehow simulate the understanding of the human mind and extract the conceptual fields of a text document, rather than focusing on the specific words used. To do so, Wordnet introduces the "senses" of each word, that is the different meaning a word can get when found in random context.

Hierarchic Structure

Furthermore, relations between words are defined which help in organizing the complex conceptual structure of a language. The most frequently encountered relation among words is the one of hyperonymy and hyponymy. This includes the conceptual relationship between for instance the words "house" and "bedroom" or "animal" and "cat". This approach at the end forms a tree which the root node "entity". Of course a hyponymy relation is transitive. A cat is a mammal, a mammal is an animal, thus a cat is an animal. Relationships are not limited in hyperonymy and hyponymy, but they extend further to meronymy, antonymy.

Further relationships

Wordnet's complexity increases as it differentiates the meaning structure for different parts of speech. Noun, verbs, adjectives and adverbs are treated not only independently, while forming

a hierarchy, but also the cross-relation between them is taken into in "morphosemantic" links, which exist between words that share the same stem with the same meaning.

Complexity

The complete description of Wordnet is out of the scope of this project. So is also the calculation of the complexity of word and sentence comparison. But from a more generic point of view, the complex structure of Wordnet and the numerous relationships defined for each pair of words translate in numerous combinations that need to be examined while comparing two text document. Thus, a Wordnet based application is quite computationally expensive and it is up to the user to define whether or not it is worth it to spend more computing power in order to acquire delayed but highly accurate results.

2.2 Frequency of Words Based Approach

The most common approach to compare two documents with respect to their similarity is to compare the appearances of the specific words in the two documents. This is also referred as a frequency of words approach, because a such algorithm would calculate how many times a word is mentioned in the documents and with the help of a scoring algorithm will assign a similarity score for the two documents.

2.2.1 Content Representation

The content is represented as shown in the figure 2.1.
All users' texts are joined to form an index. This index includes several documents. Each document is actually the activity of every users. It is shown that a document can have more than one so-called "fields". Those fields are indexed for searching. For the example of the twitter users, one field could be the tweet texts, which is the only field to be used in searches, while another field could be the date when the those tweets are posted or the id of the user. This differentiation of the fields helps so as to distinguish between fields that will be used for searches and fields that just provide complementary information.

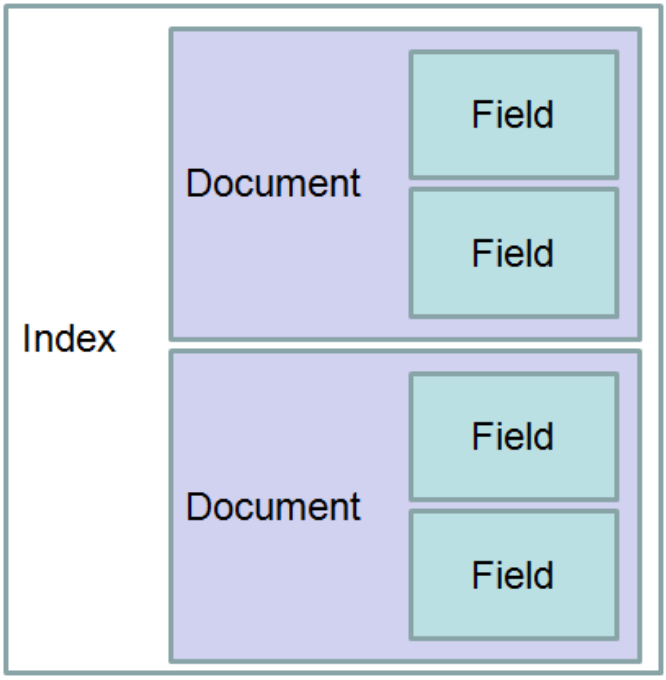

Figure 2.1: Content representation in a frequency of words based approach

It is quite often that before indexing analyzers and tokenizers are used. These two tools are almost ambiguously used in a frequency of words approached in order to reduce the size of the index and to keep only the important information concerning the texts.
Analyzers are providing the utility of eliminating all the punctuation characters from the texts before indexing. In other words dots, commas, exclamation marks etc are removed prior to indexing. A further utility of analyzers is to skip also all the common used words. Articles such as "a", "an", "the" etc, personal pronouns such as "I", "you", "he" are removed also prior to indexing. The analyzer's stopwords, i.e. words that are not considered for indexing, can be chosen and extended from the user of the program, depending on their preferences.
Tokenizers can find the roots of the words. After doing so, instead of adding in the index all the different grammatical forms of the same root, such as the verb, noun, adjective or adverb, the tokenizer keeps only one form representing the concept of the root.

An example is provided to show the use of the analyzers and tokenizers. Suppose that the following text needs to be indexed:

Author1: Konstantinos
Date of statement: 15.09.11
Statement:The investigator investigates the strange (!) case of the murders. Murdering should not be considered as a crime.

Author 2: Chaloulos
Date of statement: 16.09.11
Statement: Has criminality reached it's top here in Zuerich? Murdering is an act that needs to be investigated; or not?

In this example, not all fields concerning the statements need to be indexed for searches, at least when it is planned to do some text comparison. Thus, for searching only the field "Statement" will be indexed for searching. The analyzers will remove punctuation marks and common English words. Tokenizers will keep only the root of the words; let for our example that to be the verb format. Thus the two statements will look like:

Statement 1: *investigate strange case murder consider crime*
Statement 2: *crime reach top here Zuerich murder act need investigate*

The index will include all terms appearing after the analyzing and tokenizing of the statements. As a result the following terms will be included in the index:
Index = { investigate strange case murder consider crime reach top here Zuerich act need }

We have shown how the index is created in such approaches and how prior to indexing some tools are used so as more useful information is kept into the index. Of course the tools have no mind in order to provide judgment on whether some statement makes sense or not. The created index of the example would be the same even if the words appeared in other grammatical forms and in other order. The number of occurrences of the same roots of the words is of importance for a such approach. After explaining the data representation we see how it is possible to score a document pair depending on the similarity between the two entries.

2.2.2 Scoring Algorithm

The scoring algorithm is based in the term-frequency inverse document frequency (tf-idf) method. Term frequency correlated to the number of the appearances of a term in a document queried. The inverse document frequency correlates to the inverse of the number of the documents a term appears.
The combination of this two statistical values are used as weights in order to evaluate how important a word is to the document and to a corpus. Various implementation and theories exist on how tf and idf are used to produce the weight. In general, the idf reduces the significance of words appearing often in a large corpus. On the other hand, while examining one single document, tf is descriptive of how important one term is for this document. In designing a ranking

function to evaluate the importance of the terms in a corpus and in a document, one should take into consideration normalization factors so as to prevent a bias towards long documents, in which a term may appear much more often independently of the importance of that term in the document.

2.2.3 Lucene

Apache Lucene(TM) [8] is a high-performance, full-featured text search engine library written entirely in Java. It is a technology suitable for nearly any application that requires full-text search, especially cross-platform.
Apache Lucene is an open source project available for free download. It is maintained by the Apache Software Foundation, a non-profit organization that is consisted of software developers. These developers form a decentralized community with members from different places in the world. Lucene was created by Doug Cutting in 1997. Lucene was added to SourceForge.net under the GNU General Public License (GPL) in 2000, which was the first open source release [9].
In 2005, Lucene was a top level Apache project. It had already stated to integrate into web search engines, providing the technology behind popular sites and applications.

Lucene's scoring algorithm

In this section the scoring algorithm implemented by Lucene is presented and explained. The similarity score between two document, where one is the query document, correlates to the cosine-distance between the two documents.

Lucene is using the Vector Space Model (VSM) of Information Retrieval in order to represent the documents. This means that every document is represented as a point into a multi-dimensional space. This point is produced as the summation of many orthogonal vectors. Each of these vectors is created by an indexed term. So, a document containing several terms has values in several dimensions. The weighs of each vector are the ones that will at the end assign the similarity score between the documents. These weights are produced by an *tf-idf* based algorithm. More specifically the scoring algorithm between a document *d* and a query *q* in lucene is given by the expression 2.1:

$$score(q,d) = coord(q,d) \cdot queryNorm(q) \cdot \sum_{t\ in\ q} tf(t\ in\ d) \cdot idf(t)^2 \cdot t.getBoost() \cdot norm(t,d) \tag{2.1}$$

where t represents the terms, the *tf(t in d)* correlates to the term's frequency, *idf(t)* stands for inverse document frequency, *coord(d,q)* is a score factor based on how many of the query terms are found in the specified document *d*, *queryNorm(q)* is a normalizing factor used to make scores between queries comparable, *t.getBoost()* is a search time boost of term *t* in the query *q* as specified in the query text and *norm(t,d)* encapsulates a few boost and length factors.
Diving in more details, the *tf(t)*, which correlates to the term's frequency (the number of appearances of a term *t* in the scored document *d*), is by default given by:

$$tf(t\ in\ d) = frequency^{1/2} \tag{2.2}$$

Thus documents in which the term t appears more times are given a higher score for this term.
The *idf(t)*, which as mentioned stands for the Inverse Document Frequency, correlates to the inverse of the number of the documents in which the term *t* appears. This practically means that terms that appear rarer in the index have higher contribution to the overall score. More specifically the *idf(t)* is given by default by:

$$idf(t) = 1 + log\frac{numDocs}{docFreq + 1} \tag{2.3}$$

where *numDocs* is the total number of the documents and docFreq is the number of the documents that the term t appears in.
The term *coord(q,d)* is a score factor that depends on the number of the query terms of *q* that

are found in the scored document *d*. This translates into higher scores for documents in which more of the query's terms appear in.
A normalization in order to have comparable scores is provided by the *queryNorm(q)* factor. This factor does not influence the ranking between the documents because all documents are multiplied by the same factor. Though it makes the scores from different queries and even from different indexes comparable. The computation of *queryNorm(q)* is give by:

$$queryNorm(q) = \frac{1}{sumOfSquaredWeights^{1/2}} \tag{2.4}$$

where the *sumOfSquaredWeights* indicates the sum of squared weights of the query terms and is computed by the query Weight object of the lucene's libraries.
The *norm(t,d)* factor, which includes also a few indexing time boost and length factors, such as the *document boost*, *field boost* and *lengthNorm(field)*, has a value that is encoded as a single value before being stored. This value is retrieved by decoding at search time back to the float norm value. This coding and decoding comes with the price of losing some precision. Thus decode(encode(x)) = x is not guaranteed, while reducing the index size.

To sum up, as explained Lucene's scoring algorithm is based on the tf-idf concept. It uses some normalization factors to provide accountability and comparability of the derived similarity scores. More detailed description of the scoring algorithm can be found in the official website of lucene, under the class *similarity*.

2.3 Latent Dirichlet allocation

Another approach that can be used in order to extract some similarity between text documents is a latent Dirichlet allocation (LDA) approach. With LDA it is feasible to do some text classification. Each document is modeled as a finite mixture over an underlying set of topics. The topics are modeled as a mixture over an underlying set of topic probabilities. Those topic probabilities provide an explicit representation of a document.

A detailed and complete mathematical analysis of the LDA extends further of the score of the semester project and can be found in [10]. From a more simplified point of view, with this approach a tool that provides the following utility can be created: Texts and an array of topics of interests are given as an input on the tool and the output is a array of percentages of how much belongs the text to the respective topic.

This approach was considered as not efficient for twitter users comparison, mainly because the tweets of the users can not easily be classified into topics; this is further explained in the next chapter. Thus its functions and parameters are not presented in this project, but for the shake of completeness it is mentioned as a possible way of achieving text classification (and from that point which a scoring algorithm also text similarity and comparison).

Chapter 3

Twitter users' similarity

In order to compare two twitter users based on their text so as to extract a score of how much similar they are, metrcis need to be created. Tweet texts look quite different compared to formal newspapers' articles and web sites.
Text documents appearing in books, newspapers or in the internet are usually more structured documents with a much more clear overall content. The writers approach a specific topic and the vocabulary used is topic-relevant. Depending on the level of writers sophistication, on the targeted readers and the simplicity of the topic, the language used differs. But in all cases, the writer is interested to make his text understandable; the terminology chosen is taken from a set of words which is related to the topic of the document.

3.1 Tweets characteristics

Tweet texts appear to have some characteristics. After experimenting with tweet texts, we saw that twitter users often user the "at" character (@) to mention the name of one of their friends. Moreover, hash tags (#) where often used to express an interests on a current trend. It is also really often that they post some urls which they find interesting. Last it is also probable that they will mention the city where they are located or planning to visit.
These characteristics were the inspiration for the creation of quick and efficient metrics for inferring shared interest.

3.2 Comparing Tweets with Wordnet

The comparison between two twitter users can be implemented by both a content based approach and a frequency of words approach. Focusing on the primary case, Wordnet can provide the tools needed to complete a comparison.
There are two main reasons why the evaluation a content based technique did not take place for this project. Firstly the higher computational requirements for the many "senses" of wordnet and secondly the fact that the tweet texts look quite different from what is usually provided as input to wordnet tools.
Having to examine all senses produced by every word in the documents to be compared increases the number of comparisons needed to be made. Furthermore, the relations between words need to be taken into account transforming a wordnet comparison tool as a slow tool to work with.
When taking a closer look at how tweet texts look like, one can realise that looks like a collection of randomly selected sentences. Twitter users do not analyze a specific topic as it is the case in book texts or newspapers' articles. Users everyday experience something random in their lives and they post about it. Thus it is more usual that they will include limited vocabulary and the words will not be so much correlated to each other. A content based approach a highly accurate method [11], with correlation to human judgment approaching 70-80 %. The main reason for such high scores is that the data sets from which the texts to be compared is taken, either include carefully chosen word pairs, so that it is easy for a human to realise whether in these pairs

the words have any kind of relation, or are articles from newspare or interent [12]. In the latter case, only some of the senses of the words contained in the documents to be compared match with the senses of the other words, thus it is easier for an algorithm to realise which senses to take into account. The terminology used in the texts is topic related and the scores between different documents is far more distinguishable. In order to make this statement more clear, a simple example is provided. Imagive two text documents; one reporting the progress of a football game and one discussing about the political strategy of a goverment. It is quite probable that the word "goal" can appear in both texts. For a content based approach, "goal" in the sport report will take the sense of the football ending in the net of a team, while for the politic article the word "goal" will take the sense of a target to be achieved. The two distinguished senses of the shame word provide better results when the comparison needs to be done, especially after including further topic-specific terminology. On the other hand, a twitter users can be mentioning the same words in diffferent context on a tweet by tweet basis. Depending on whether in one day he watched a football match or he just followed a political discussion at television.
Concluding a content based approach was not preffered as a candidate for extracting a similarity score between two twitter users. Thus we state clearly that it would be quite interesting to use such a technique in tweet texts so that the intuitively made conclusion matches (or not) with some real evaluation results.

3.3 Comparing Tweets with Lucene

A frequency of words based approach was chosen as a tool to extract a similarity score between twitter users. Lucene's libraries are used in order to provide the framework on which a program is developed.

3.3.1 Tweet Similarity Metrics

Instead of indexing for searches all the terms of the tweet texts, the special characteristics of the texts are used to provide some quicker results. Five tweet features are used to create compare tweets and create metrics by keeping only some of the terms appearing inthe tweet texts. Terms kept and indexed are:

1. the terms after the "@" character, that is the names of the people mentioned by the users
2. the terms after the "#" hash tags, which are the current trends
3. any cities mentioned by the user
4. the terms after "www." or "http://", which are the urls posted by the users
5. all the above terms

3.3.2 Discussion on the Metrics

In this section a further discussion on the metrics defined is presented. We provide a self critique on whether they are representative measures of sharred interest and what should be expected from each metric.
The "mentions metric", which is the one keeping the terms appearing after the "@" characters, is expected to give only few matches. It is more representative of the "communication" cycle of the triangle presented into the introduction chapter. It somehow provided information on who want to communicate with whom. Thus, this metric could be helpful in deepen the understanding of the relation between the interconnection of the "interest" cycle with the communication cycle.
The "hash tags metric", which is the one keeping the terms appearing after the "#" characters, is expected to be the most strong and real interest sharring similarity metric. This is mainly because the act of tagging something inheretly means having interest in it. As a result two users tagging the same topic matches them as similar. This metric is purely a metric to measure the sharred interest between users.

The "cities metric", which is the metric keeping only the cities mentioned by users, is expected to provide insight on how people move. Thus it could be helpful in deepening the understanding between the "interest" cycle and the "mobility" cycle of the triangle presented in the introduction chapter.
The "url metric", which is the one keeping only the terms appearing after the "www." and "http://" sequences, is more representative of the sharred interest with respect to internet activity and posting urls. Unfortunately it is not possible to extract from the domain name of the url useful information concerning the similarity on the posts. This is mainly because twitter has introduces tools to shorten the links because there exists a 140-letter limitation in the posts. Thus, this metric provides insight on two users' similarity towrards posting urls.
The "combination metric", which is the one which indexes all the above terms, is the most complete metric. It is expected be the most reliable metric since it includes the most terms. It provides similarity scores taking into consideration information about who wants to communicate with whom, what are the trends a user follows, location information and internet preferences.
Including less terms into the index speeds up the comparison between the users. These is because less terms are indexed, thus less comparisons needs to be made. Moreover, the terms used are of some importance to the users, compared to common words of the English vocabulary which might be very commonly used. If those common words had be also included into the index, it would be more probable that two users, who in a conseptual level share no interests, would be scored as a highly similar pair. For instance, only two uses could be using the word "nice". For a human mind it is more understandable that this word does not provide vital information in extracting sharred interest between the two users. With this frequency of word based approach, "nice" would have high inverse document frequency and term frequency for the two users. Thus the pair would be scored as similar. Towards the direction of avoiding this kind of biasing, keeping only specific terms can be a solution.
The tradeoff of course is that less information is included into the index. The fact that the terms indexed are of some real importance for the users does not exclude the existance of other words that are also of importance and can also describe the users' preferences.

Chapter 4

Methodology

4.1 Tools

In this section, the software and data that were needed to complete the project is presented. Initially a "mysql" server [13] was needed in order to extract the tweet texts from a .dump file containing several information about tweet users. Two databases, one from Gowalla and one from Twitter were provided. The two databases concern the same set of users and thus allow for correlation between the Gowalla "check-ins" and the twitter texts, from which the shared interest can be inferred.

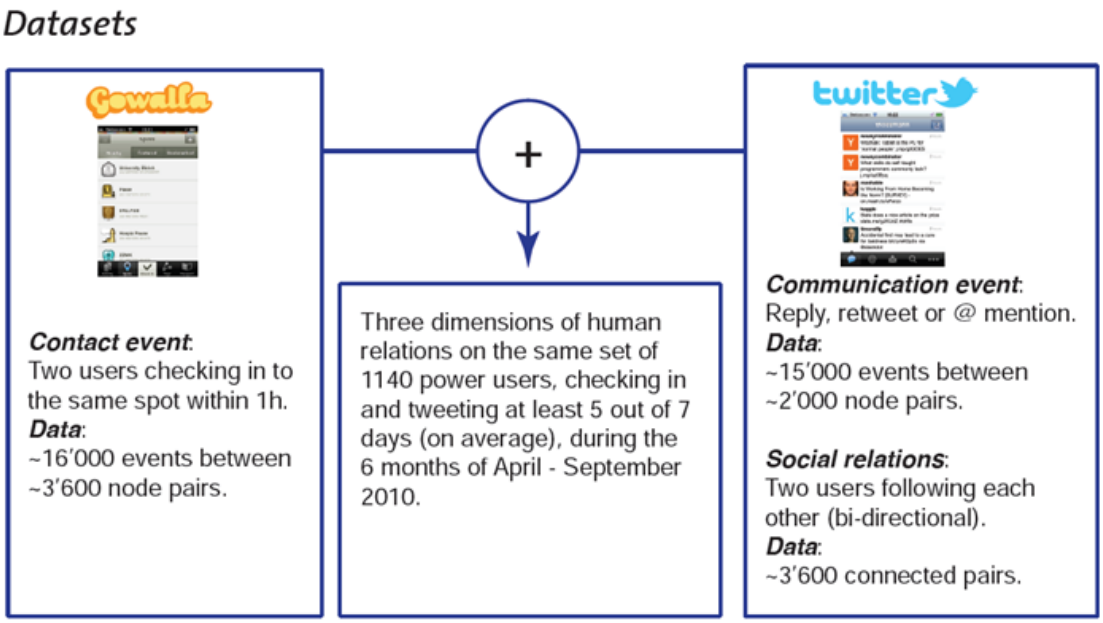

Figure 4.1: Gowalla and Twitter databases

The texts from randomly selected users are saved in .txt files. For the thesis a program written in java is created. This program is based on the open sourced Lucene libraries and extened so as to take as an input those .txt files and extract similarity scores between each pair of users. The indexing, text processing and design of similarity metric and scoring are all done by this java written program.
For the metric concerning the cities mentioned by the users, a database including all cities of the world is download from [14]. Again with the help of "mysql" those cities are extracted in a text file. Each word of the resulted text file is queried in the documents to be compared. If a match is found, then this term is included into the index. After the text comparison, the similarity scores produced by the Lucene's scoring algorithm are saved in matrices.

Matlab [15] was used for processing those matrices and producing plots and the cross correlation.

4.2 Experiment

The limited memory and computational power available for experiments restricted the number of users and tweets that are used for evaluating the five metrics. 250 twitter users are randomly selected and 250 tweets of those are kept. After the comparison is done, five similarity matrices are produced for the five metrics.
The histograms and the complementary cumulative distribution functions for all the metrics are plotted at the figures 4.2 - 4.9.

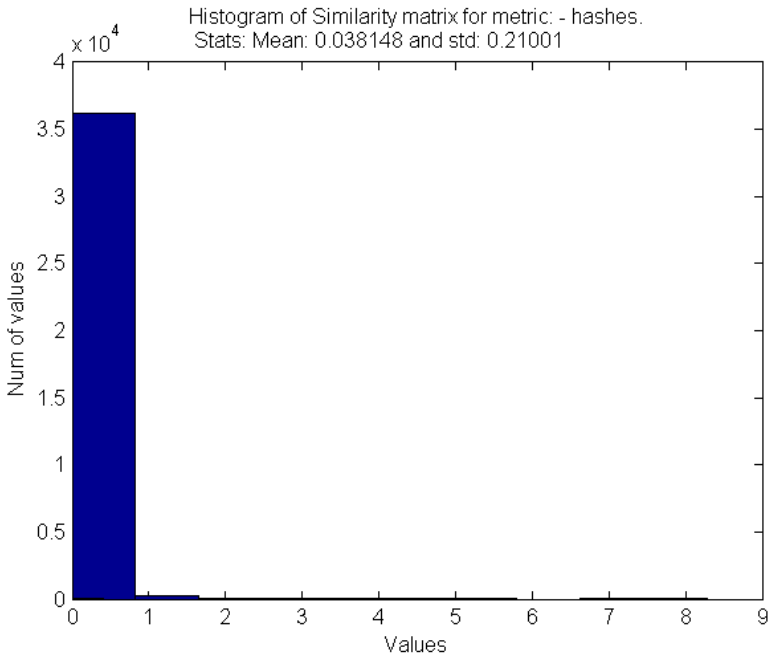

Figure 4.2: The histogram of hashes metric scores.

For the set of users tested, the cities metric did not find any match between the users and thus figures for this metric are not to be presented.
The histogram of the scores resulted from the hashes metric are shown in the figure 4.2.

The percentage of the values that are bigger than the 0.1 of the highest value of the hashes metric is 0.85%. That can be translated in few very strong matches and many values close to zero. Although it looks like this metric can not provide good results, we focus the reader's attention to the size of the test set. For only 250 users it is reasonable that few matches would be found. Moreover, this metric is purely a interest-oriented metric, since in a way the twitter users declare the name of the topic that they want to comment on. Thus, a match of this metric could be alone an efficient and reliable metric for inferring shared interest between twitter users.

The histogram of the scores resulted from the url metric are shown in the figure 4.3.
The url differentiates significantly. The percentage of the values that are bigger than the 0.1 of the highest value of the url metric is 9.77%. This means the url metric assigned more average scores and found more matches between the users. This is quite expected, when one considers that twitter users, who are also internet users, are much more likely that they will be interested into sharing informations concerning their web activity.

The histogram of the scores resulted from the mentions metric are shown in the figure 4.4.

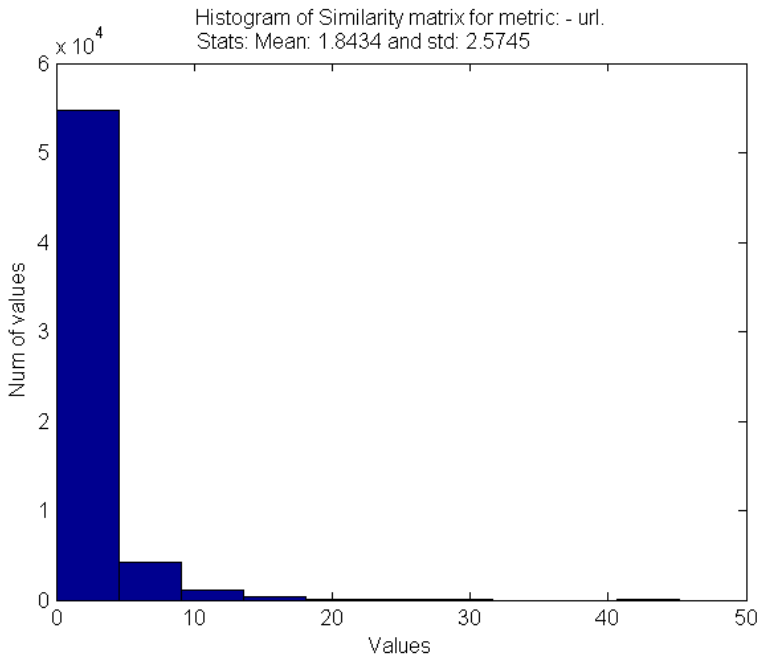

Figure 4.3: The histogram of url metric scores.

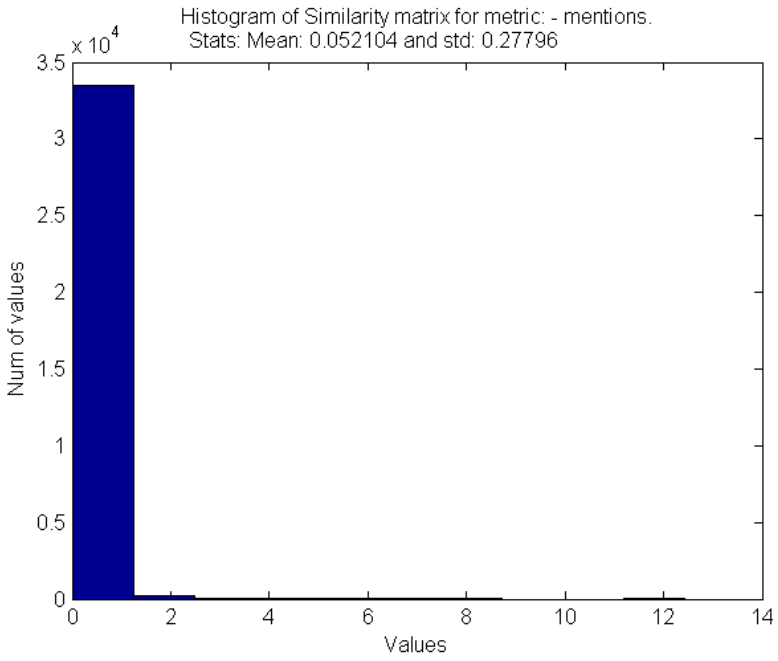

Figure 4.4: The histogram of mentions metric scores.

For the mentions metric, the percentage of the values that are bigger than the 0.1 of the highest value of the mentions metric is 0.71%. This means even fewer very strong matches and more values close to zero, compared to the hashes metric.

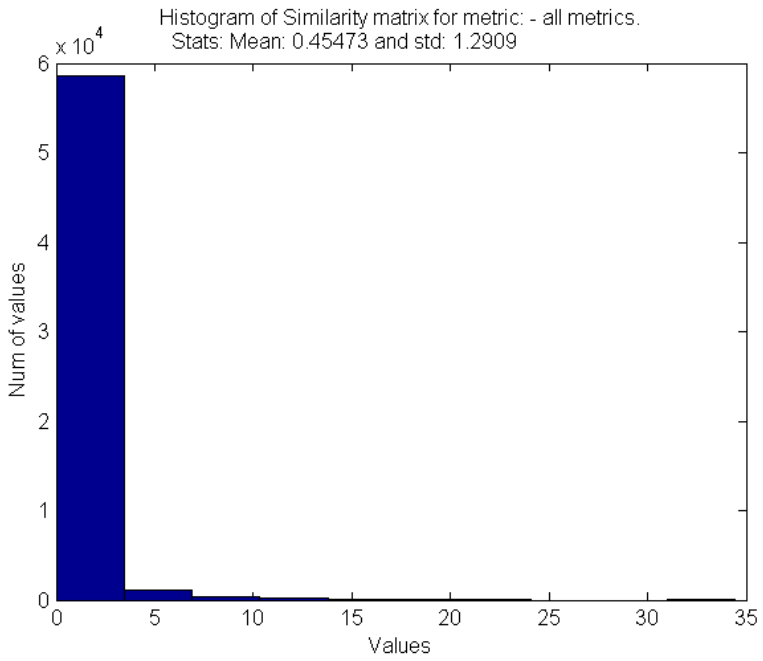

Figure 4.5: The histogram of the combined metric scores.

The histogram of the scores resulted from the combined metric are shown in the figure 4.5.

While comparing the complementary cumulative distribution functions of all the metrics, it can be derived that indeed the combined metric gives the most complete results by scoring pairs with more moderate scores, giving values in most of the range of scoring.

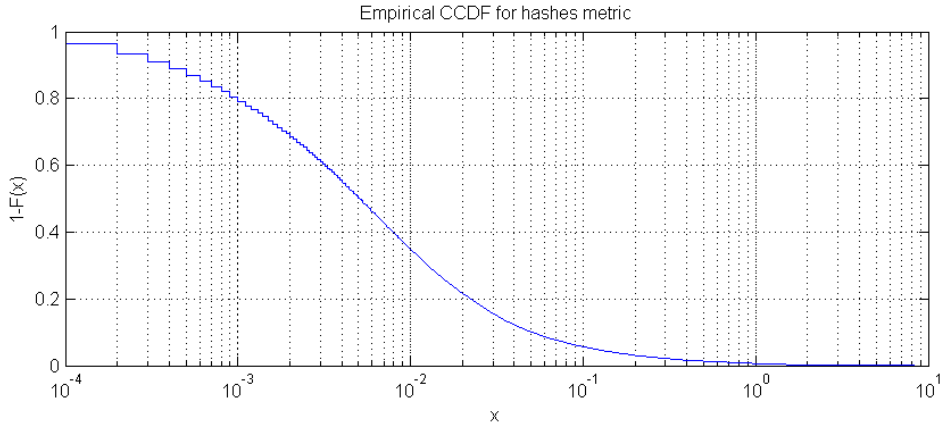

Figure 4.6: CCDF of hashes metric.

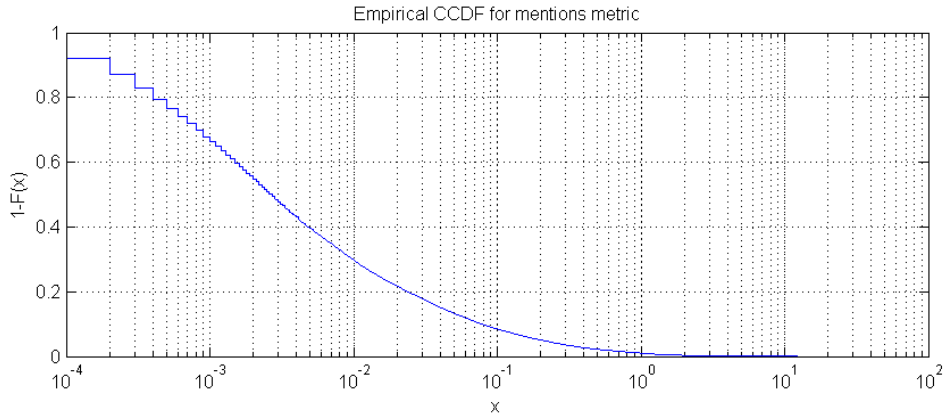

Figure 4.7: CCDF of mentions metric.

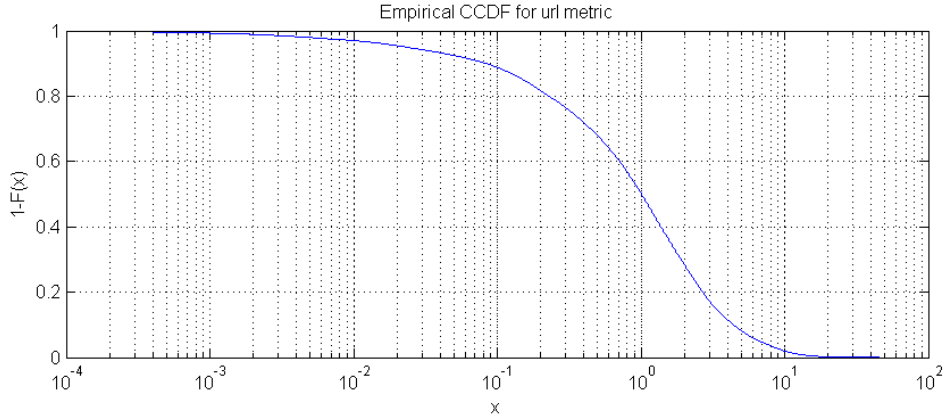

Figure 4.8: CCDF of url metric.

The combined metric is expected to be somewhere in between. It includes all the terms indexed by the other metrics. Thus, the absence of matches for the hashes and mentions metrics is compensated by the matches found in the url metric. Similarly, the high scores of the url metrics are decreased, because the importance of a match of the urls metric is decrease in the combined metric, which includes more terms. So we see that the percentage of the values that are bigger than the 0.1 of the highest value of the combined metric is 2.96%.

Metrics	hashes metric	mentions metric	urls metric	combined metric
hashes metric	1	0.3073	0.0815	0.1104
mentions metric	0.3073	1	0.2038	0.2820
urls metric	0.0815	0.2038	1	0.7856
combined metric	0.1104	0.2820	0.7856	1

Table 4.1: Pearson's correlation between metrics.

In table 4.1 it is shown the correlation produced by the Pearson's product-moment correlation coefficient. This coefficient is widely used to show the strength of linear dependence between variables.
From the table 4.1 it is observed that the combined metric, which is the most complete metric since it includes the most terms, is higher correlated with the urls metric. This practically means that the importance of the urls metric's matches is the highest for the overall similarity score, mainly because the user set used for experiments contained many urls. This result allows the

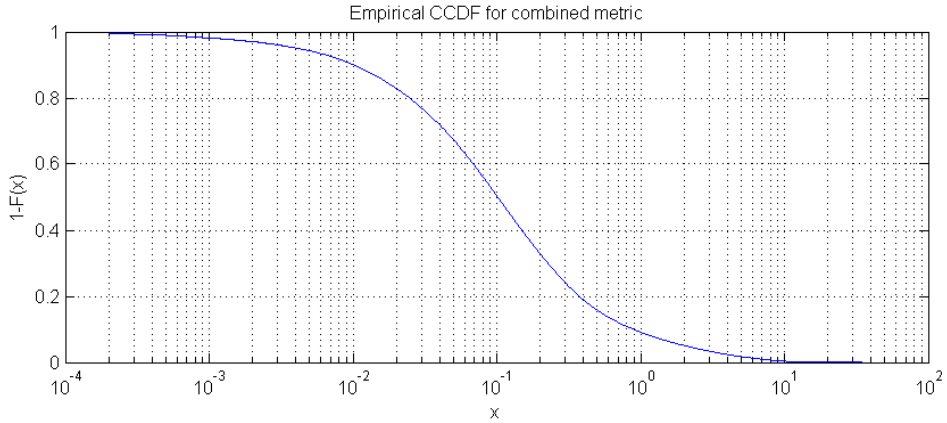

Figure 4.9: CCDF of combined metric.

urls metric to be used as a representative metric, but in the same time less computational expensive, for extracting shared interest between two twitter users. Of course one should notice that the similarity scores resulted from this metric would concern merely the similarity of the user pairs with respect to the World Wide Web.

The absence of matches for the cities metric does not automatically mean that this metric should be thrown away. For bigger experiment sets it will definitely find some matches and assign scores on the level of location similarity.
The results presented concern a really small data set. This comes from the fact that 250 randomly selected twitter users are a really tiny portion of the total twitter population, which approaches 200 million users in 2011 [16], who generate 65 million Tweets a day [17].

Chapter 5

Future Work

This chapter is included to provide some possible extensions of the current work, as well as goals that were wished to be achieved.
As a starting point, perhaps the most significant task to be performed, which adds credibility to any similarity metric designed, is to do a ground truth comparison of the produced results. In other words, comparing the similarity scores produced by an algorithm to human assigned scores. Although human judgment is not always the so mentioned ground truth, it is widely the most accepted method to realistically evaluate the similarity between texts.
Though we would like to draw the attention on the difficulty on judging tweet texts. As explained in the text comparison chapter, tweet texts consist of random posted sentences. Thus even a human mind cannot extract which information may or may not be vital for a user. The process of comparing two users were tested by some volunteers. It was anonymously admitted that the procedure of scoring pairs, apart from the tiring and uninteresting reading of the tweet texts, was difficult to be done. It is quite difficult to decide on which level the comparison should be done. Sure the texts can be similar in many ways, but it was quite difficult to realize whether some shared interests existed or not.
Furthermore, the comparison with human assigned scores was infeasible for completion as a part of a semester project. The high demand on human resources requires further planning of how the comparison can be done efficiently, so that volunteers do not waste a lot of time, as well as how the set of available judges can be increased (with the use of a web interface platform).
A second suggestion for future work is to do a combination of content based and frequency of words approach for text comparison.

Bibliography

[1] B. Distl, G. Csucs, S. Trifunovic, F. Legendre, and C. Anastasiades, "Extending the reach of online social networks to opportunistic networks with PodNet", MobiOpp, 2010

[2] A.-K. Pietilaeinen, E. Oliver, J. LeBrun, G. Varghese, and C. Diot, "Mobiclique: Middleware for mobile social networking", in WOSN, 2009

[3] A. Mtibaa, A. Chaintreau, J. LeBrun, E. Oliver, A.-K. Pietilaeinen, and C. Diot, "Are you moved by your social network application?" in WOSN, 2008

[4] N. Eagle, A. S. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data", PNAS, 2009

[5] T. Hossmann, T. Spyropoulos, F. Legendre, "Stubml: Using Facebook to Collect Rich Datasets for Opportunistic Networking Research", AOC 2011

[6] T. Hossmann, F. Legendre, T. Spyropoulos: "Analysis of Three Dimensions of Human Relations: Mobility, Social and Communication Interactions" Poster, Applications of Network Theory, 2011

[7] Wordnet website, *http://wordnet.princeton.edu/*,
20-09-2011

[8] Lucene website, *http://lucene.apache.org/*,
20-09-2011

[9] Gospodnetic, Otis; Hatcher, Erik. 2010. "Lucene in Action, Second Edition", *http://www.manning.com/hatcher3/*,
20-09-2011

[10] D.M. Blei, A.Y. Ng, M.I. Jordan "Latent Dirichlet Allocation", Journal of Machine Learning Research 3 (2003), 993-1022

[11] Alexander Budanitsky and Graeme Hirst: "Semantic distance inWordNet: An experimental, application-oriented evaluation of five measures" Dep. of CS, Un. of Toronto

[12] G.Varelas, E.Voutsakis, P.Raftopoulou "Semantic Similarity Methods in WordNet and their Application to Information Retrieval on the Web", WIDM'05,Bremen

[13] Mysql website, *http://www.mysql.com/*,
21-09-2011

[14] MaxMind website, *http://www.maxmind.com/app/worldcities*,
21-09-2011

[15] MathWorks website, *http://www.mathworks.ch/products/matlab/index.html*,
21-09-2011

[16] BBC news website, *http://www.bbc.co.uk/news/business-12889048*,
26-09-2011

[17] Your world, more connected, *http://blog.twitter.com/2011/08/your-world-more-connected.html*,
26-09-2011

[18] Extending the reach of online social networks to opportunistic networks with PodNet,
A B. Distl, G. Csucs, S. Trifunovic, F. Legendre, and C. Anastasiades, MobiOpp, 2010.

[19] Title of the relevant work2,
Author/Company that published the work, Place and date of publication,
www.relevantwork.com,
Date of your last visit (if URL)

[20] Title of the relevant work3,
Author/Company that published the work, Place and date of publication,
www.relevantwork.com,
Date of your last visit (if URL)