Correlating Semantics and Expertise to Enhance Social Network Exploration

David Vogt

A dissertation submitted to the University of Dublin

in partial fulfilment of the requirements for the degree of

Master of Science in Computer Science

University of Dublin, Trinity College

2011

Declaration

I declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

_____

David Vogt, 30 August 2011

Permission to lend and/or copy

I agree that Trinity College Library may lend or copy this dissertation upon request.

_____

David Vogt, 30 August 2011

To my parents

Acknowledgements

I am very grateful to my supervisor Dr. Owen Conlan for sharing his knowledge and supporting me throughout the course of this dissertation. His concise feedback, many interesting discussions and his patience allowed me to have enough creative freedom while maintaining a clear focus to carry out this research.

I also would like to thank Dr. Alex O'Connor for participating in many of the discussions and for making very valuable contributions to this work.

Finally, I would like to thank all participants of the evaluation and the developers of the Alchemy API for allowing me to perform a sentiment analysis based on their technology.

Correlating Semantics and Expertise to Enhance Social Network Exploration

David Vogt

Supervisor: Dr. Owen Conlan

Assistant Supervisor: Dr. Alex O'Connor

University of Dublin, Trinity College, 2011

Abstract

Social Networking has witnessed substantial growth in recent years. However, with this growth comes an increasingly complex level of connectedness and a large number of multi-threaded communications that users can find overwhelming. The potential and the richness of Social Networking are compromised by this burden which often affects the end-user's ability to identify relevant information. To address this problem, this research is proposing to use unstructured data from Social Networking and correlate it with structured information from other sources, such as the Linked Data initiative. It is envisaged that the combination of expert-enhanced annotations of a domain and the utilisation of social signals further improve the quality of selected entries. In addition, a tool to create and manage a knowledge-domain in order to build tailored views on Social Networks will be developed. Over 2.8 million entries from Twitter have been downloaded and used to create and evaluate these tailored views. Additionally, the sentiment of the collected data is used to create and evaluate variations of the developed algorithms. This research will empower individuals and organisations to deeply understand information derived from many users and to traverse through large amounts of data in a more efficient matter.

Table of Contents

List of Tables

List of Figures

List of Source Code Details

1 Introduction

1.1 Motivation for Research

The success of Social Networks has increased rapidly in recent years. Facebook currently has more than 750 million active users, with 50% logging on in any given day. There are over 900 million objects that people interact with (pages, groups, events and community pages). More than 30 billion pieces of content (web links, news stories, blog posts, notes, photo albums) are shared each month. In total, people spend over 700 billion minutes per month on Facebook[1]. Twitters user base is 175 million and 95 million Tweets are created every day[2].

These statistics illustrate the immense amount of data produced and the very high degree of connectedness between users and entities, as well as among users themselves. The vast amount of information that exists in Social Networks has the potential to allow users and organisations to access extremely valuable data. Despite this potential, Social Media content often is not utilised because of the very same reason: the quantity makes it seem nearly impossible to keep up with the pace of data being published and to extract the important pieces of content. Effectively leveraging this information and making a distinction between noise and relevant content has become a significant challenge and is putting a burden on users and organisations.

Twitter and Facebook are two examples of currently very popular websites. A general abstraction can be made that common characteristics of these and other successful projects are often the sharing of very short pieces of information, commonly referred to as "Microblogging". This form of communication is very likely to spread further in the future, since the internet will cover more and more areas of live and the success of these projects has proven that people have a desire to express themselves online. Ordinary blog-entries in the past years usually consisted of at least a number of paragraphs of text. This nature of information provides a lot of context and it is relatively easy to develop a system to search

[1] http://www.facebook.com/press/info.php?statistics
[2] http://twitter.com/about

blogs for keywords or other patterns, see how they are related, and then identify if an article relates to a certain topic.

Microblogging however raises a number of new problems:

- Conversations are complex and multithreaded
- Often jargon and abbreviations are used
- Massive amounts of data are being created and change dynamically
- There is very little semantic information about an entry and a lot of implicit knowledge

Other users are assumed to understand the meaning of an entry through other context such as the history of the authors' entries, previous conversations or other background knowledge. One must know about many bits of information and context within the Social Network as well as distributed on other sources.

Developing a system to help with the process of leveraging this information would create the possibility to deeply understand content derived from many users. This is also a significant opportunity for organisations and companies: People express opinions, which, if properly aggregated and interpreted, can be used for a variety of use cases, such as the analysis of market opportunities, political insight or feedback on products.

1.2 Hypothesis

The proposed research will include the investigation in the semantic analysis of unstructured pieces of data and how this information can be correlated with other sources of information such as the Linked Data initiative in order to discover relevant pieces of content. By tailoring the identified information an engaging and meaningful view on Social Media data can be created.

Investigating in user-interfaces that allow a user or an expert to create its own tailored view on Social Media content is part of this research. An expert may be a brand manager who has detailed knowledge about the domain, while a casual user may just have personal preferences and a desire to specify what content he is interested in to simplify the process of gathering relevant information.

The different signals in Social Networks provide an opportunity to select relevant information. This research addresses the question of how these signals relate and which of

them have the biggest impact on relevance. It is envisaged that expert-enhanced annotations of a domain can support the process of identifying relevant information. Further, it will be investigated if the sentiment of the content has an impact on the perceived relevance. This work raises the question to what degree can sentiment analysis, social signals, the utilisation of external semantics and expertise improve the selection of relevant Social Media content.

1.3 Main Research Aims

To address the proposed hypothesis, the following main research goals are set for this work:

- To develop a system to retrieve Social Media data, to cache it and to make it accessible in an efficient matter
- To investigate in tools and techniques to identify the subject of short pieces of content and to reduce noise, spam, duplicates and partial duplicates
- To use open knowledge initiatives, such as Linked Data, to help categorise unstructured content and to identify related keywords to a domain
- To identify the signals which have the biggest impact of the relevance of such entries
- To investigate in tools to determine the sentiment of the collected data and to measure their impact on the relevance
- To visualise identified pieces of information and to evaluate the quality of the presented data

1.4 Technical Approach

To reduce noise of Social Media, a method to determine the relevance of pieces of information must be developed. Characteristics of possible relevance of an entry within a Social Network exist in today's most common projects:

In Twitter, the persons followed, persons who use similar hash-tags, persons that "Retweet" (redistribute) entries of the author or persons that the user had a discussion with can be seen as more relevant to the user. Besides these directly correlated characteristics to the user, nondependent features of "Tweets" have an impact on relevance: Popularity of the author of an entry (How many followers? How many "Re-

Tweets”?); Timeline (Recent or past entries?); Trending Topics (How many people talk about it? What is the acceleration of the number of posts about a topic?).

In Facebook, the algorithm to create an evidence of relevance may use different attributes, such as: Number of interactions (Messages, Wall posts, Chats); Number of pieces of information in common (Friends, Pictures, Interests); Social structure (Family, Partner, Group memberships, Site subscriptions); Popularity of the entry itself (Comments, number of “Liked it”); Timeline (Recent or past entry?); Preferences and interests of the user (Number of clicks on entries of a certain user? Defined priorities through friend lists? Selection/Removement of certain users on News Feed?).

After careful investigation in related work, attributes will be chosen to design algorithms to select content from Social Networks. The algorithms will be enhanced through a correlation with other sources of information and by expert annotations. These algorithms will be evaluated against a set of users to analyse if a pattern of relevance can be identified. Further, a system to semantically describe a domain of interest will be developed and evaluated. A data corpus which allows an efficient, flexible and broad exploration of Social Media content of a domain will be built and cached by using state of the art technologies.

1.5 Outline

Chapter 2 will give an overview of related work and background information in the field of Social Network analysis, metadata, open knowledge initiatives and natural language processing. In chapter 3, the design of the system and the various components will be discussed. The implementation will be presented in chapter 4, including technical challenges and an in-depth discussion of the algorithms used to select relevant data from Social Networks. In chapter 5, the metrics and methodologies of the evaluation will be highlighted. Further, the results of the conducted survey will be presented and analysed. Finally, chapter 6 concludes the findings and proposes further research.

2 Related Work

2.1 Overview

In this section related work and similar technologies will be discussed. First, approaches to create semantic Social Networks and the general architecture of such systems will be evaluated. A special focus is put on Twitter since this research will be using it as a source for Social Media data; therefore a thorough analysis of its characteristics will be necessary. Projects trying to predict trends and to extract and enrich unstructured data will be examined. Further, vocabularies and open knowledge initiatives will be crucial to gain a better understanding of unstructured data; therefore Linked Data and DBpedia will be discussed. Finally, basic approaches to perform a sentiment analysis will be presented.

2.2 Semantics of Social Networks

2.2.1 The Future of Social Networks

In a paper on the future of Social Networks [1] the authors predict a general movement to the vision of semantic Social Networking, which envisages that Social Networking concepts are built into the fabric of the next-generation internet itself.

Current projects often have a problem with the number of connections between their users and the meaning of connections: Sometimes an increased number of “friends” exists just to make a user look more popular. The strength about the connection is not quantified and the list of contacts may function more like an address book. This does not show what relationships actually mean and which relationship is valued more than another.

As a result there is a challenge of extracting the strength of connections between people. This problem has been investigated by other research, e.g. by taking into account how often people correspond or how similar users are, based on their network and behaviour [2].

Another challenge is the possible existence of multiple Social Networks for a user: In different projects a person needs to re-create his social graph, find and add interests and friends again. The authors propose that people and things should be connected in an interoperable extensible way. For instance, multiple projects can communicate and one

global unique Id (e.g. OpenID[3]) is used to identify a person spanning over multiple networks [1].

To achieve these interoperable communications, projects such as FOAF[4] to describe relationships among humans and SIOC[5], standardising how entries in blogs and forums are accessed, have been proposed.

It is envisaged that these semantic Social Networks will change the entire experience of the internet and used for many other, indirect purposes: E-Mails could be filtered based on the relationship between author and recipients and relevant websites can be identified by analysing the reputation of a website in one's Social Network. In future systems, social algorithms may decide the routing of a message to relevant recipients. For this research, the move to semantic Social Networks is important because of the richness of these networks: Instead of looking at just one project and correlating its data with structured data, the detail gained out of connections between many projects would provide more context and semantic meaning and simplify the interpretation of data derived from many users.

2.2.2 The Flink System – A Semantic Social Network

This projected [3] investigated the idea of building a Social Network bottom-up: Looking at varied sources of information and creating the Social Network based on existing connections. The authors argued that a Social Network does not necessarily have to be defined actively through its users, but can be derived from the way people use existing communication technologies.

The project chooses a particular user group: Researchers in the field of the Semantic Web who participated or held and organisational role in one of the International Web Conferences or Semantic Web Working Symposiums.

The data was first acquired from the different sources of Information: E-Mail Mailinglists, Google Scholar (for references and co-authorship), the Friend-of-a-Friend Project and the regular Google Page (to identify cross-relations between two given topics).

[3] http://openid.net
[4] http://www.foaf-project.org
[5] http://sioc-project.org

The collected data then was enriched through a number of techniques: e.g. Identify-reasoning based on E-Mail Addresses or geographic lookups based on the address of a person. For the visualisation, the JUNG Framework[6] (a universal network and graph framework) has been used.

One particular interesting part of this work is the usage of search engines to reason about correlations between topics. This is a crucial requirement for this proposed research: The relation between two entities could be discussed by investigating in structured data such as DBpedia. Another approach would be to query Google for the entities both separately and together in order to take the number of results as an indicator for its correlation.

The project visualised how people are connected and how strong their connection is, it also showed details about a single person derived from different sources of information. Another interesting result was a map with all the topics that the community talks about, and how these topics relate to each other.

When concluding, the authors identified one problem which will also have a great impact on this research: All dimensions of the Social Graph where represented as one single graph. People might be talking about different topics on different technologies. A strong connection between two researchers could either be based on a lot of collaboration in the research area or on frequent communications with another technology (e.g. a mailing list).

In Social Networks such as Twitter, users talk about many different topics potentially with many different groups of people. These groups can be seen as different dimensions of the Social Graph: Family, Co-Workers and people with the same interests and friends. There is a high potential for creating different visualisations for different dimensions of this graph.

2.2.3 Citizen Sensing, Social Signals, and Enriching Human Experience

The authors of this work propose that humans could be seen as sensors to better understand the meaning of a situation [4]. Humans are better at contextualizing and discriminating data. For instance, regular cameras just capture everything and in later steps computer systems face the challenge of evaluating the data. Humans decide what is important naturally, they can perform complex reasoning and the collected information will be of a much higher quality than the one taken by regular sensors.

[6] http://jung.sourceforge.net/

A citizen sensor network is defined by an interconnected network of people who actively observe, report, collect, analyse and disseminate information via text, audio or video messages, through various sensors such as GPS and cameras.

Microblogging projects such as Twitter sites have strongly encouraged the emergence of citizen journalism in a way that was not predictable. In their work, the authors cite the example of Boston police which allows people to report suspicious events and people by their phone. The police then uses this information to control their units and prevent crimes. For this proposed research, a user could be seen as a sensor based on his location. The weighing of a piece of information can be affected by his location (e.g. if a user is reporting from a demonstration he might have more valuable information than somebody who is geographically far away). After collecting the sensor data it can be semantically annotated and correlated. The location of a message can be used to find the address and prominent locations nearby. In succession, other projects (e.g. Flickr or Wikipedia) can be utilised to find similar content.

The project Twitris is one possible implementation of the theory of Citizen Sensor Data [5].

2.2.4 Characteristics of Twitter and the Nature of its Data and Usage

In this section an analysis based on the crawling of the entire Twitter site and after retrieving 41.7 million user profiles, 1.47 billion social relations and 106 million Tweets will be discussed [6]. The goal was to create a better understanding of how Twitter is used and to find influential people by ranking users by the number of followers and by Google Page Rank.

The authors identified that over 85% of trending topics are headline news or persistent news in nature, which proves the high correlation of Twitter and real events. The dynamics of Retweets showed that any retweeted entry reaches an average of 1.000 users, regardless of how many people followed the author of the original Tweet. Retweets almost instantly reach the 1st, 2nd, 3rd and 4th hop (distance to original author) in the network.

The first step was to collect the data on which the analysis will be performed on:

- User Profiles: Starting with a particular popular user ("Perez Hilton") a breadth-first search along the direction of followers was executed. Twitter allows only a limit of 20.000 queries per hour from one host-machine, based on its IP-Address. To increase

this rate, 20 machines with different IP-Addresses and self-regulating download speeds were used. Additionally, profiles of those who refer to trending topics in their Tweets during the time of the experiments where collected.

- Trending Topics: The top 10 trending topics where captured every five minutes via the Twitter Search API[7].
- Tweets itself: All Tweets that mention trending topics where collected. Twitter allows a maximum of 1,500 Tweets per query; the related entries were downloaded every 5 minutes. The collected data included the full text, author, written time, ISO standard language code, receiver if the entry is a reply and a third party application if available.
- Spam removal: A mechanism of the Firefox add-on "Clean Tweets"[8] was used to filter entries with a high potential of spam. It removes Tweets from users who have been on Twitter for less than one day or have more than three trending topics.

The main Results of the experiment were the following:

- The majority of people with less than 10 followers never tweeted or did just once.
- The Median number of Tweets posted stays flat up to 5.000 followers and then grows with magnitude.
- Only 40 users have more than a million followers; these users are mostly celebrities and news organisations.
- Reciprocity: 77.9% of users are connected one-way (not following in both directions) and 67.6% are not followed by any of their followings in Twitter. Other projects are connected much more bi-directional and this illustrates that Twitter often is used more as a source of information than an actual communication platform.
- Degree of separation: The average connection path length (hops) from a user to any other given users is 4.12 which is surprisingly small for a not very bi-directional network.

Trending topics:

- Keywords from Google Trend[9] were correlated with Twitter trends based on substrings (if 70% of the keyword is the same it is considered a match). Only 3.6% out of 3,479

[7] http://search.twitter.com/api
[8] http://www.seoq.com/blvdstatus/clean-tweets.html
[9] http://www.google.com/trends

unique trending topics from Twitter existed in 4,597 hot topics in Google; Google and CNN news headlines were ahead of time with hot topics compared to Twitter.

- User Participation: About 20% of Twitter users participate in trending topics through creating content.

Besides these statistics, a number of important papers on the dynamics of Twitter exist. For instance, the influence of users and the impact of their social status on actual influence have been discussed [7].

2.3 Identifying Trends and Analysing Social Networks

2.3.1 Identifying Trends through Semantic Social Network Analysis

The goal of this research was to perform a Social Network analysis based on algorithms for mining the web in order to identify trends and people who start trends [8]. The analysis was based on the system "Condor" (formerly TeCFlow [9]), a software for predictive search and analysis of the internet and Social Networks.

The general motivation was to prove that in today's internet economy a buzz about a topic on the web reflects its popularity in the real world; hence events in the real world might be predictable through an analysis of the internet and Social Networks.

The authors took a concept from a particular domain (e.g. US presidential elections), mapped it to online Social Networks and the web buzz index and made estimations of how the trend might develop in the future.

Three different information spheres where crawled: The web at large ("crowds", the largest set of data, e.g. news sites or company sites), the blogosphere (experts) and forums (discussions/swarms).

A significant difference when looking at the web at large, including news sources and public sites is that it is up-to-date with the real world. Some Social Networks lag behind and consist rather of responses to topics. The question arises if these responses are more accurate than the news sources itself when trying to identify the perception about a topic in the world.

The gathering of the data and its analysis was performed in three steps: What (identify concepts and their relative importance in the information sphere); Who (actors using the concept); How (positive or negative sentiment).

What:

- The authors used the betweenness, widely used in graph theory, to describe the centrality of a concept: An approximation of its influence on the discussion in general (numerical between 0 and 1)
- To calculate the betweenness, Google and Google Blog Search results on a topic were used: The top 10 results for a topic are the 1st level, then the 10 pages pointing to each of the results of the 1st level are the 2nd level, this is iterated until the desired level of granularity is reached. The more central sites are referencing a page, the higher is the betweenness.
- The importance of an individual concept depends on the linking structure of the temporal network compared to other concepts.

Who:

- What certain people say carries more weight (a person can be a website or a blog); therefore the individual importance of a website should be considered. For instance, if a website has a general high betweenness it will increase the betweenness of the concepts when it is covering it.
- "Kingmakers" are websites which act in this way; they can be seen as more important than websites that have a low betweenness individually.

How:

- Finally, the sentiment of the identified information has to be analysed (positive, negative or neutral)
- The system used a simple word lists for positive/negative words, stop words (the, and) and for irrelevant words (see 2.6 for details and possible approaches).
- Regular expressions were used to find co-occurrences of terms, e.g. a company and keywords in a piece of content (e.g. Goldman, Goldmansachs, Goldman, Sachs)

To verify the results, the authors correlated the crawled data and its hot topics with actual proven data. For instance, the development of stock market values was compared to the buzz on the web about these companies and products, which successfully showed a clear correlation.

One general problem of the analysis was noise. In the example of the presidential candidates, most of the estimations where correct except of one outlier: Al Gore had a higher betweenness than Hilary Clinton, even though the votes clearly favoured Hilary Clinton. This was caused by the popularity of Al Gore due to his movie release "An Inconvenience Truth", which happened to be released at around the same time as the election and it caused a lot of discussions and review on the web. Al Gore was interpreted as a politician and as a producer and actor of a popular movie. Similar to the work presented in section 2.2.2, this again addressed to problem of mixing up multiple social graphs which are not directly related.

Overall, the results of this project were sophisticated, though it is questionable if a system should have such a strong dependence on the ranking results of Google, where there is no public insight about how websites are ranked. Other search engines, such as Blekko[10], give an insight on how a ranking actually was created.

2.3.2 YouTube Comment Analysis

In this work, dependencies and correlations between comments, views, comment rating and topic categories of the project YouTube[11] have been analysed [10]. YouTube is a video-sharing platform and one of the most visited websites worldwide. It allows users to comment and rate videos. In addition, videos can be grouped in categories and channels.

The research goal was to predict community acceptance for unrated comments based on their identified sentiment and to build classifiers for the estimation of a comment rating.

The authors addressed the following core questions:

- Can community feedback for comments be predicted?
- Is there a correlation between the sentiment of a comment and the rating of a comment?
- Is the rating of a comment an indicator for polarizing content?

[10] http://www.blekko.com
[11] http://www.youtube.com

For the experiments, a sample of six million comments out of 67,000 YouTube videos has been gathered, including the voting of a comment ("Thumb up" or "Thumb Down"). Over 50% of all voting's had no voting or where neutral (same amount of positive and negative votes). The other part of the distribution has a tendency to be positive.

The overall prediction was that the selection of words used when composing a comment will have an impact on the acceptance of the comment in the community. The sentiment of a comment is evaluated by using SentiWordNet[12], a lexical resource for opinion mining. Additionally, a generated list of 50 positive and 50 negative words based on positive and negative comments and the number of occurrences of the word was created when trying to predict the rating of a comment. Words like "music", "love" or "best" were accepted as positive, "idiot", "kill" or "loser" where negative. Supervised learning paradigms were used to train a classification model based on Naïve Bayes, naturally leading to better results with a higher number of training data.

Further, the authors identified polarizing YouTube content by examining the variance of comment ratings for each video. The estimated polarization (polarizing, rather neutral, in between) was correlated against a manual rating. This manual rating was collected from tree test-persons who were asked to rank the perceived polarization of 100 videos. The results showed a very high agreement which proved that a high variance on comment ratings indicate polarizing content.

Finally, the category in which a video resides and the impact on the comment ratings was examined. Some categories such as "Music" have much more positive comment ratings than others (e.g. "Gaming" or "Science"). The assumption was made that there might be more spammers and younger people in categories such as "Gaming", which leads to negative comments.

The challenge for this research is that in Social Networks often there is no numeric rating available. Finding the polarizing topics will be extremely valuable; other patterns besides the variance of ratings will have to be used when working with Social Media from Twitter.

[12] http://sentiwordnet.isti.cnr.it

2.3.3 Twarql – Semantically Querying Twitter

The goal of this research was to translate entries from Twitter to Linked open Data[13] in real time [11]. The following steps were performed in order to retrieve and annotate the data:

1. Extract the content (e.g. based on entity mentions, hash tags or URLs)
2. Encode content in RDF using shared and well-known vocabularies (such as. FOAF and SIOC[14])
3. Enable structured querying of Microposts with SPARQL
4. Enable subscriptions to a stream of Microposts that match a given query
5. Enable scalable real-time delivery of streaming annotated data

Details about the extraction of the data, which is based on the most current semantic-web technologies, can be found in an earlier paper on “Linked Open Social Signals” [12]. To solve the problem of the vast amount of entries in Twitter, this project only looked at a subset of entries which has been downloaded between two points in time.

After the raw data is received it is run through the Twarql system in order to add sentiment and a correlation with DBpedia entities. After the extraction, Tweets are encoded in RDF, using the ontology stacks FOAF and other vocabularies. A full-fledged query language can be used to query the data (SPARQL), which is much more expressive than just using keywords of Twitter.

This system could be used for brand tracking; the authors gave a demonstration on getting the number of posts about keywords and the sentiment on the touchscreen device “Apple IPad”. Based on SPARQL, queries such as “List all URLs that people recommend with relation to my product" or “List all people that have said negative things about my product" can be asked.

2.4 Open Knowledge Initiatives and DBpedia

For this research, open knowledge initiatives will be used to achieve a better understanding of Social Media. These projects aim to create a standardized language of describing content in order to allow interlinking and dynamic querying. One popular project is Linked Data which will be discussed in the following subsection. Further, DBpedia is a community effort

[13] http://linkeddata.org
[14] http://sioc-project.org

to extract information from Wikipedia, which will be of interest in order to help annotating a knowledge domain.

2.4.1 Linked Data

Linked data is an approach to publish structured data in a standardized way to make it reusable, more connected and generally more useful and machine readable. Motivated by Tim Berners-Lee, the director of the World Wide Web Consortium, the following design principles to publish data on the web have been defined[15]:

1. Use URIs (Uniform Resource Identifier) as names for things
2. Use HTTP URIs so that people can look up those names
3. When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL)
4. Include links to other URIs so that they can discover more things.

Examples of datasets include DBpedia, FOAF profiles and Project Gutenberg. The last publication of data includes 19,562,409,691 triples (datasets)[16].

2.4.2 DBpedia: A Nucleus for a Web of Open Data

DBpedia is one of the most prominent datasets of Linked Data, representing big parts of the Wikipedia[17] encyclopaedia [13]. An entity is identified by its name in Wikipedia, which has the advantage of community census about a term: A wide range of encyclopaedic topics are defined by community members and their agreement on it, clear policies for their management exist.

The project can be used as a general repository to refer to “things” and make them uniquely identifiable. The data can be accessed in three ways:

- Through Linked Data
- SPARQL, an endpoint for clients to query the data, often useful when a developer exactly knows what is needed
- Downloadable RDF dumps, useful if larger parts of the data are needed

[15] http://www.w3.org/DesignIssues/LinkedData.html
[16] http://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/DataSets/Statistics
[17] http://www.wikipedia.org

Rich queries can be performed against the DBpedia dataset. For instance, the DBpedia Faceted Search[18] allows a user to create filters and search for complex questions such as "Rivers that flow into the Rhine and are longer than 50 kilometres". To achieve this, desired properties in the triples are defined (e.g. item type: River, river mouth: Rhine, length (m) 50000 and up).

In this research, querying this sophisticated data source based on criteria received from Twitter and other sources will allow a better understanding of the context of a piece of information. Further it will help to discover related information and connections to other datasets.

2.5 Personalisation and Adaptive Systems

Relevant work has been done in the field of spanning complex and semantically meaningful queries across separate data sources [14]. This can be used as a starting point for semantically enriching unstructured data through other sources of information.

Generally, there has been a movement in personalisation towards model driven and service oriented approaches. Significant work towards the adaptive-web and methods of personalisation has been proposed by Peter Brusilovsky [15] [16].

In order to allow a personalised browsing through complex linked data derived from Social Media, a new way of exploring this information will be necessary to reduce noise. An interesting approach to this problem has been introduced by Melike S. et Al. [17], who proposed a system which adds a semantic layer to the web by using Linked Data and adaptive hypermedia. As a result, the user receives a better guidance through the web and receives personalised, context-aware links to other relevant information.

Another project introduced a third party adaptive service which allows a user to have a unified cross-website personalised experience [18]. This work is relevant in order to support the proposed research when building a personalised experience for distributed sources of information.

[18] http://wiki.dbpedia.org/FacetedSearch

2.6 Natural Language Processing and Sentiment Analysis

Sentiment analysis focuses on the problem of classifying documents and pieces of information by sentiment, e.g. if a review or Tweet is positive or negative overall. This will be a crucial success factor for this proposed research in order to understand the meaning of large amounts of data derived from many users. Sentiment analysis usually returns a certain degree of probability and for this work the focus lies on the separation between “positive”, “negative” and “neutral”, where “neutral” can be interpreted as not distinguishable.

Sentiment analysis should not be confused with topical categorization, which is the attempt to automatically assign topics to pieces of information. This also is an important part of this research which will be approached by correlating the unstructured pieces of information with structured data.

Topic-based classification is using keywords to find out about the meaning of a piece of information, sentiment analysis often faces the challenge that there is so no clear keyword or combination of keywords and therefore more understanding of the language is needed.

One particular problem the authors of a system to analyse the reviews of movie titles identified is the filtering of sarcastic comments [19]: Often a user would start describing a list of all his expectations and things that should be good about a movie and then negate it in the very last sentence of his review.

Another challenge comes from the high use of special slang, grammar rules and demographic differences. Sometimes entities are named with English words, e.g. the album name “Music” by Madonna or Stephen Kings book “It”, which also makes it difficult to gather the correct entity and sentiment.

To run the experiments, data of movie reviews was used and three machine learning methods where employed: Naïve Bayes, maximum entropy classification and support vector machines. Sample data (only negative or positive) from the Internet Movie Database

(IMDb[19]) was used as input. This data was chosen because both the textual rating and a "star-based" rating exist which make it easy to verify the extracted sentiment.

Two students choose positive and negative words that they associate with a rating of a movie. The range of accuracy of correct sentiment detection was 58-64% but a lot of ties (number of bad and good words equal) occurred. In the next step, instead of using the defined list by students, frequency word-counts were performed on the comments with known sentiment. This led to a positive and negative word list which classified only 16% of the comments as ties. Words such as "?" were not picked by humans but led to much better accuracy, arguably because questions and confusion about a film tends to indicate a negative perception. The general observation is that humans are not very good at picking keywords for sentiment analysis.

To increase the performance and simplicity the authors replaced words like "didn't" with "NOT_DID", so matching algorithms could be developed with less complexity. Machine learning methods surpassed all random accuracies. Some attempts of an optimisation via Bigrams to get more context and also through looking at the position of the keyword (e.g. assume that movie review starts with sentiment) has been analysed. The result of these attempts led to minor improvements but could not increase the result significantly.

2.7 Summary of Related Work

In this chapter, related projects and required technologies to semantically correlate Social Networks with other sources of information in order to select relevant data have been presented. Characteristics of Twitter, its nature of communications and usage as well as general statistics were illustrated. Projects that use sources such as blogs, forums and news sites to identify trends and perceptions about a topic were discussed in section 2.3. The work on YouTube comment prediction presented the impact of polarising content on the acceptance of new comments in the community. Open Knowledge Initiatives and DBpedia in order to create a connected, semantic web have been illustrated in section 2.4. Finally, trends in personalisation and adaptive systems as well as in sentiment analysis have been discussed.

[19] http://www.imdb.com/

3 System Design

3.1 Overview

In this chapter, the key concepts and the approach to address the research question of chapter 1, resulting in the design of the “Peregrine”-system, will be discussed. Functional requirements as well as an initial analysis of the domain will be developed. The process of annotating a domain in order to build a data corpus of relevant entries will be discussed. Further, different algorithms will be designed in order to evaluate which signals are the strongest indicators for relevance. Finally, the key concepts for the user-interface in order to support these tasks will be described.

3.2 Requirements

To define the requirements it is important to distinguish Peregrine from related work. It is envisaged to create a system to gather, rank and display Social Media based on semantics and expert-enhanced annotations. Further, social signals and the sentiment of an entry will be utilised to create variations of the selected entries.

Other work [8] aimed to identify trends through ranking entries based on third-party technologies such as Google PageRank. This work is aiming to rank entries based on defined semantics and social signals. Rather than identifying trends, the goal is to find the most relevant entries to match the defined semantics and to reduce the noise of Social Networks. It is important to have a clear insight on how entries are selected; therefore no remote technologies will be used for rankings in this research.

Semantics will need to be used and stored in order to annotate a domain and to receive data. DBpedia [13] and the Linked Data project [20] provide a framework of how technologies such as RDF and SPARQL can be utilised to dynamically store and query data and to make it universally accessible. This work will utilise these technologies but is not aiming to create new datasets for Linked Data or to build new knowledge about the storage of complex semantics. Unlike the Twarql [11] [12] project, this work is not trying to build an end-point for semantically querying Social Networks.

Sentiment analysis will be part of this work and will be an attribute for the evaluation. While the development of a classifier specifically tailored for this work might lead to the

best results, it is envisaged that a third party provider with basic sentiment analysis functionality will provide results sufficient for this work. Building a classifier, e.g. based on the concepts of related work [19], is outside of the scope of this research.

To address the problems identified in chapter 1 and after considering the related work from chapter 2, the Peregrine system will need to support the following operations:

- Expert-Enhanced selection of products, related keywords and authors of interest
- The crawling of Twitter based on the defined semantics
- Sentiment analysis on obtained data
- Visualisation of the collected data ranked by different signals

In addition, the developed user interfaces need to provide basic usability: Casual users should be able to traverse through the presented data without difficulties; experts should be able to create and annotate a domain of interest efficiently.

3.3 Architectural View

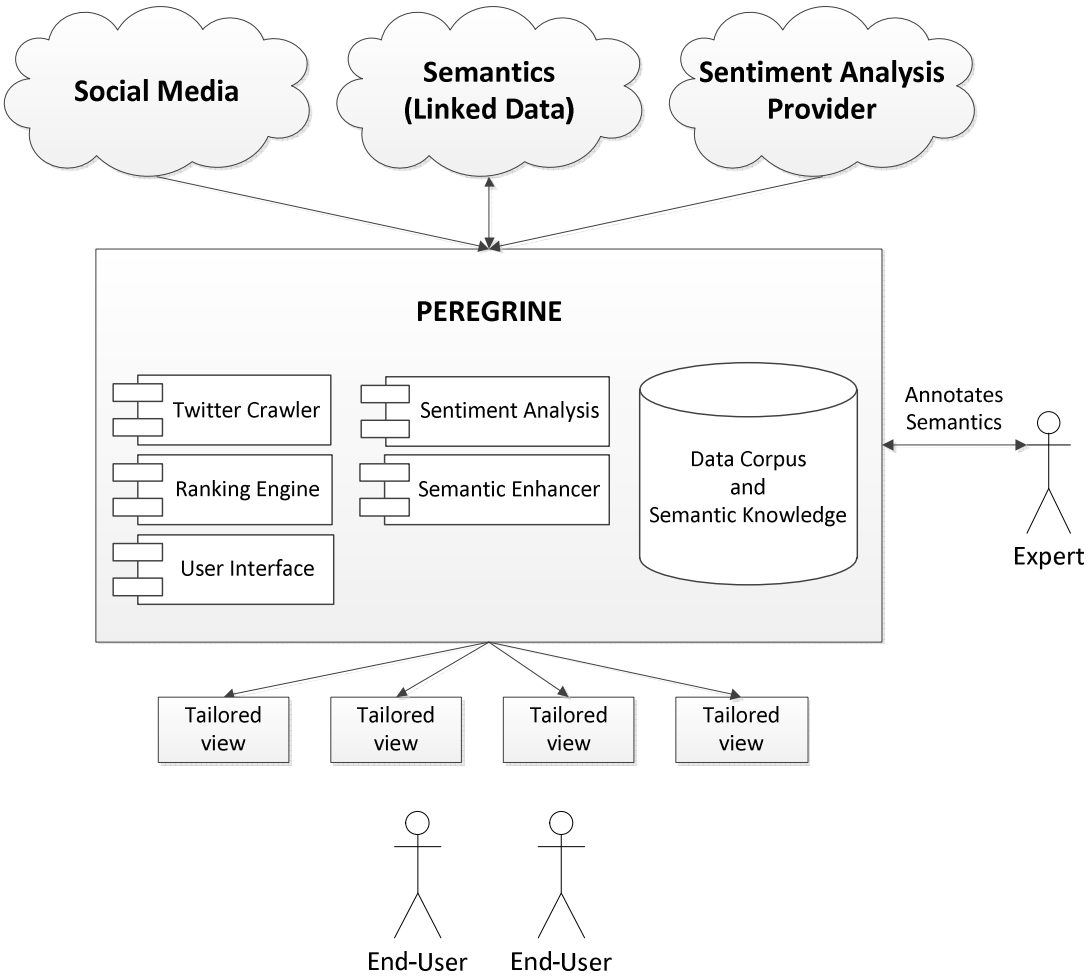

Figure 1 Architectural View

Figure 1 illustrates a high-level architecture and the components of the Peregrine system:

- *Semantic Enhancer:* External semantics are used to allow an expert the annotation of a domain
- *Twitter Crawler:* The annotations are used to receive data from Twitter
- *Sentiment Analysis:* The collected data is analysed and a sentiment score is assigned
- *Ranking Engine:* The entries are ranked based on the expert-enhanced semantics, social signals and sentiment results
- *User Interface:* The selected Social Media data is displayed in tailored views

3.4 Initial Analysis of Suitable Domains

3.4.1 Choosing a Domain of Interest

It became apparent that it is necessary to find a compelling use case to visualise a knowledge domain. In order to identify an interesting domain, it is important to understand what topics are being actively discussed in Social Networks. A prototype implementation of a java-based client to download a random selection Tweets, by using the "sample-stream" of the Twitter API[20] (a one percent sample of all Tweets worldwide), was developed for this purpose. The main reason why Twitter was chosen is the nature of very open communications. Unlike other prominent Social Networks such as Facebook, created data in Twitter is public by default, which allows the gathering of a lot of data in a short period of time.

The goal of the initial crawl was to determine popular topics and to generate an estimate of how much data can be obtained from Twitter. The following steps were executed (implementation details will follow in the next chapter):

```
For each received entry
    If(language of entry = English)
        For each word,
            if not seen before
                Add to List
            else
                Increase counter of word by one
```

Source Code 1 Pseudo code for initial Twitter crawling

Roughly 600,000 Tweets where downloaded and 944,000 individual words where identified and counted. As expected, the list was led by common words such as "have" (30,495 occurrences), "I,m" (29,645 occurrences) or "not" (26,160) occurrences. A manual screen of the table led to the conclusion that some of the most discussed terms were related to technology companies, specifically Apple[21] products and related applications and accessories. With the requirement to have a lot of data and to be able to obtain data from other sources, Apple products were an appealing option to analyse. They are widely

[20] https://dev.twitter.com/docs/streaming-api/methods
[21] http://www.apple.com

discussed by professional users as well as regular consumers and they are highly documented in other sources such as DBpedia[22] and public news sources. As a result, the domain of interest was defined to products by the company Apple Inc.

3.4.2 Identifying Products and Related Keywords

To gain more knowledge and meta-information about Apple products and to create suggestions for related keywords, two approaches were considered:

1. Utilise DBpedia and fields such as “see also” or “category”
2. Utilise commercial sources that are likely to contain data about products, such as the Amazon API[23] or BestBuy API[24]

After investigation and the development of prototype-crawlers it became clear that both approaches provide useful information, but the step of filtering data from commercial APIs proved to be more difficult than using the knowledge base of DBpedia. For instance, when looking for “Apple products” it was desired to receive a list of unique products rather than a list of all products, including their individual model numbers and different versions (e.g. different hardware specifications). This information may be of interest in some cases but in this research this was not the main focus. Also, when utilising the commercial API of BestBuy, the results included many accessories which also were not easy to identify and to filter because there is no consistent signal marking them as accessories.

DBpedia offered a list of unique products and additional information when looking for a company. In the case of Apple, the following products were listed: Macintosh, IPod_Touch, IPod_Nano, IOS_(Apple), IPod, ILife, IPad, IWork, Time_Capsule_(Apple), Mac_OS_X, AirPort, Apple_Cinema_Display, MacBook_Air, Mac_Mini, Apple_TV, MacBook_Pro, IPhone and IPod_Classic.

This is a comprehensive list of products with a lot of additional information available on the DBpedia page. Additional keywords to these products can easily be gained from DBpedia. These additional keywords will be used to make suggestions to the expert when annotating a domain of interest.

[22] http://dbpedia.org/page/Apple_Inc.
[23] https://affiliate-program.amazon.com/gp/advertising/api/detail/main.html
[24] http://bbyopen.com/developer

Overall, DBpedia was the most suitable source for products and related keywords to support the aims of this work. The expert will need to be involved in order to annotate the data and to remove or add products and keywords as needed.

3.4.3 Correlation with Sources of User Reviews (Side Note)

Initially it was planned to contrast data gained from Social Networks with data gained from user review sites, such as Amazon or BestBuy. An initial implementation and evaluation with the BestBuy API was performed. When querying BestBuy for products and reviews with the term “IPhone”, it returned a comprehensive list of the product and different model variations. Additionally, about 30% of the entries consisted of accessories and related products which, as stated earlier, could not be filtered out with a very high accuracy. This problem not only existed with BestBuy but occurred in most review sites: When searching for products, the list always contained related products, an effect not desired for this experiment. A possible solution would be to include the expert and to create manual mappings between the product entity (from DBpedia) and related products (from BestBuy, Amazon or another source). This approach is not very scalable and would involve a lot of manual work which is not feasible. While the additional product version could be an enhancement of the user’s knowledge (e.g. to find out that the product is available in different colours), the additional information led to an overflow of information and to a distraction of the core goal of this research: to identify and rank signals in order to select relevant information related to a product from Social Networks.

3.5 Semantic Enhancer

It is envisaged that an expert can create a broad semantic description of a domain, consisting of products, related keywords and trusted authors. To achieve this, Peregrine will first propose products of interest of which the expert can select a set. Afterwards, DBpedia will be queried and the related keywords will be pulled. This information has to be stored persistently in a database. Further, the expert can create a set of authors of which he thinks are trustworthy, particularly interesting or influencers of the domain. Authors are domain-wide while related keywords a tied to the products. This design choice has been made because the related keywords can vary significantly from product to product while many authors cover a wide range of products (for instance macrumors.com, which discusses many topics such as device releases, operating system news and Apple related applications). To provide feedback and to simplify the adding of authors, feedback about

the author's popularity and other Twitter-related attributes should be provided at the time of adding.

3.6 Twitter Crawler

After running the initial analysis of suitable domains, it became clear that there will be delay between annotating a domain and analysing the results. Twitter allows the tracking of certain keywords or authors in real-time, however accessing past entries has some restrictions. If the user-id is known, 3200 past entries from the specified users can be collected[25]. The Twitter search API allows accessing up to 1500 entries sorted in various ways. For this work it is desired to collect a significantly higher amount of data and running multiple searches is not particularly suited for this purpose. Some other projects exist to access historic Twitter data, but for this work it was desired to use the official source and to utilise recently published information. The most appealing option was to use the filter stream[26] because it allows the long-term tracking of keywords of interest.

The result of this constraint was that the expert-interface will be tested separately to the end-user interface. The participants will not be able to see the results of their defined keywords but will evaluate the data collected over a longer period before the evaluation. The semantics to collect the content for this evaluation will be defined by a separate expert.

Only the collected data corpus will be used for the future experiments, real-time updates that qualify and could be added to the data-corpus while running the experiment will not be considered. Further, the collected data should only consist of entries in the English language.

The crawler requires a set of keywords which should be tracked on Twitter. The basic requirement of an entry to qualify is the occurrence of the product name and one additional keyword (more details on the algorithms will be presented in the next section). However, the crawler will track only the product names. This will lead to entries being collected even though they do not mention at least one related keyword at the time of the setup. The advantage of this approach is that keywords could be changed at the time of performing the experiments without the need to collect further data from Twitter.

[25] https://dev.twitter.com/docs/api/1/get/statuses/user_timeline
[26] https://dev.twitter.com/docs/streaming-api/methods

3.7 Ranking Engine

3.7.1 A Definition of Relevance

Algorithms need to be designed in order to answer the core research questions of this work: What is the strongest signal when selecting relevant Social Media content? To address this question, the term “relevant” needs to be clearly defined. In the context of this work, an entry is relevant when it improves the knowledge about a product without the need to have further background knowledge. In the evaluation it will become clear that this definition of relevance is highly biased and depends on the task a user is doing.

3.7.2 Assumptions

The following assumptions were made based on well-known dynamics of Social Networks and on former research [6] [7]:

1. Random Tweets show little relevance
2. Social status (e.g. high number of followers or Retweets) increases the relevance
3. Expert-Enhanced selections of authors increase the relevance
4. A combination of social status and an expert-enhanced selection of trusted authors may lead to the highest overall relevance

3.7.3 Algorithm Design

To qualify, a Tweet needs to contain at least the product name and one related keyword (3.4.2 showed how these product names and keyword suggestions are being gathered). Mentioning more keywords does not have an effect of the ranking of the Tweet because authors often mention many keywords or trending topics just to appear in Twitters search results or in other search engines that are looking for these keywords.

Duplicates and Spam should be identified and considered. One approach which will be implemented in this work is to consider the date of the creation of the entry and the registration date of the author (as presented in 2.2.4). To detect duplicates, a suitable algorithm needs to be implemented which not only detects exact duplicates but also partial duplicates. This is crucial because often Tweets just vary by a few characters (e.g. an additional hash tag or a minor change in a URL). For the end-user, these entries contain the same information and should be filtered out. How the Levenshtein distance algorithm helped to overcome this challenge will be presented in the next chapter in more detail. In

the following subsections the design of four algorithms to select and rank Social Media content will be presented.

3.7.3.1 Random

Entries that meet the basic qualification, sorted randomly. In addition, the selected Tweet must not be a duplicate or partial duplicate. This is the simplest algorithm without the use of any signals.

3.7.3.2 Social Status

Two social signals have a big impact on the influence of the author in Twitter: The number of followers and the number Retweets. As proposed in former research [7], a high number of followers does not always indicate that the user is very influential. However, the TwitterAPI has a limitation that does not give the exact number of Retweets of an entry when it is higher than 100. For more than 100 Retweets, the API always returns 101, which does not allow a powerful distinction to a high degree of detail. Therefore, Tweets from authors with many followers are ranked as the highest in this algorithm. A further restriction is that the author of the entry is not part of the expert-enhanced set of authors (presented in the following two algorithms), in order to separate more clearly and to avoid duplicate entries.

3.7.3.3 Expert-Enhanced

The Expert-Enhanced algorithm is only considering qualified Tweets from a set of trusted authors defined by the expert. In this work, this set consists of 39 sources that are well known technology blogs, news sites or individual authors (see Appendix A). The results are displayed in random order.

3.7.3.4 Combined

In this algorithm, only Tweets from authors defined by the expert will be considered, identically as in the Expert-Enhanced algorithm. In addition, the entries are ranked by the number of followers of the author descending.

3.8 Sentiment Analysis

To measure the impact of the sentiment on the perceived relevance of an entry, two possible approaches were evaluated:

1. Development of a separate algorithm which uses the sentiment of the entries as a signal to rank Tweets
2. Usage of the sentiment on top of the previously defined algorithms and measurement of its impact on the perceived quality overall

While the first approach would be a powerful separation to only focus on the impact of the sentiment, it would also increase the number of algorithms to evaluate for the user. The four defined algorithms seemed like a lot of data to traverse already and it was not desired to overload the user with too much data. Also, the sentiment of an entry is a somewhat artificially created signal. It would be very feasible to create a simple classifier and to ask participants to rank presented entries as positive, negative or neutral and to use this training data in order to classify future entries. However, this approach would lead to at least two required user experiments: Firstly, collect the data to train a sentiment classifier, secondly, ask the participants about their perceived relevance of the data. This approach would go beyond the scope of this research and therefore approach two will be pursued by utilising a third party service which provides basic sentiment analysis on short pieces of text.

Often the sentiment analysis on a piece of information results in a score from -1 (negative), 0 (neutral) to +1 (positive). While a more detailed segmentation might provide interesting insights, this also would increase the number of required experiments and the burden on the participants. For this research there will be no segmentation beyond these three sentiment types.

The design question of how to use the sentiment to enhance the Peregrine system in a meaningful way resulted in the hypothesis that polarising content maybe of more interest than neutral content. Negative or positive entries could be an indicator for the expression of an opinion while neutral might often be less relevant. To evaluate this hypothesis, the sentiment will be used on top of all algorithms to allow two different setups:

1. First Setup: Entries will be displayed regardless of their sentiment
2. Second Setup: Only positive or negative entries will be displayed

This results in two evaluations of the same system but with differences in which entries will be presented.

3.9 User Interface

To interact with Peregrine and to define the semantics, a user interface is necessary. Due to the described technical limitations, defining semantics and the evaluation the results are two different tasks, therefore two different user interfaces will be created. The expert-interface will support the selection of products and the management of related keywords and authors. The end-user interface will only present the selected entries based on the four different algorithms.

3.9.1 Expert Interface

The expert interface will be split of in two separate steps. First, the products of interest can be selected. Figure 2 illustrates a wireframe of this step: The Products will be pulled from DBpedia and presented to the expert who can select the relevant items.

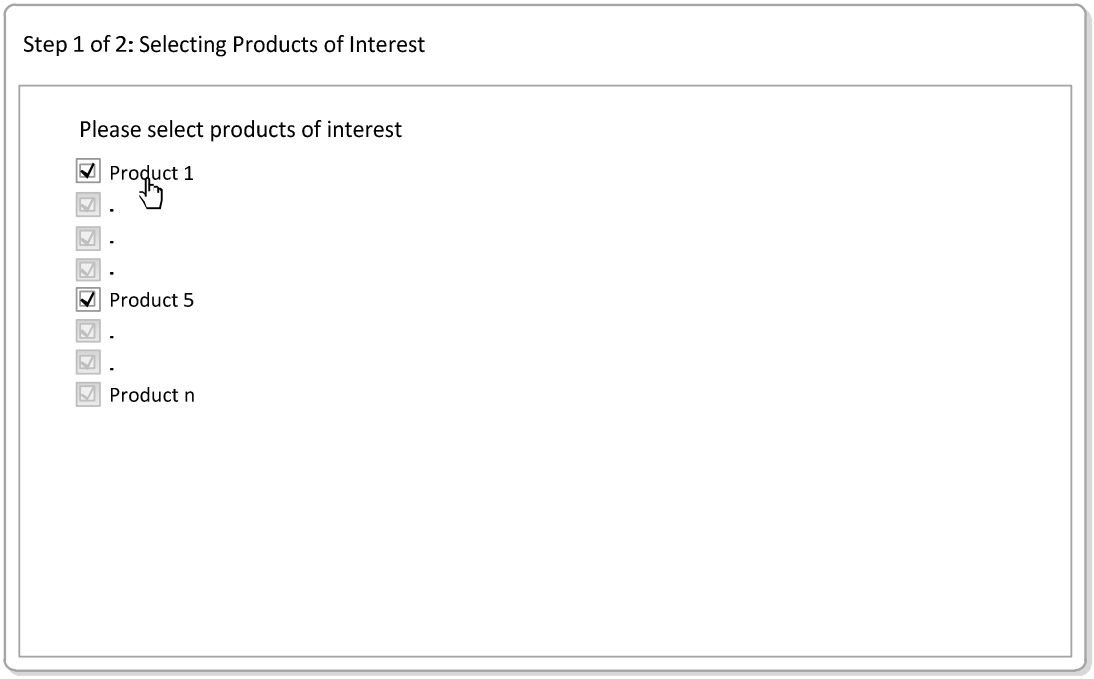

Figure 2 Expert Interface: Select products (1 and 5 selected)

After selecting the products, the second step is to annotate the domain with authors of interest and related keywords for each product.

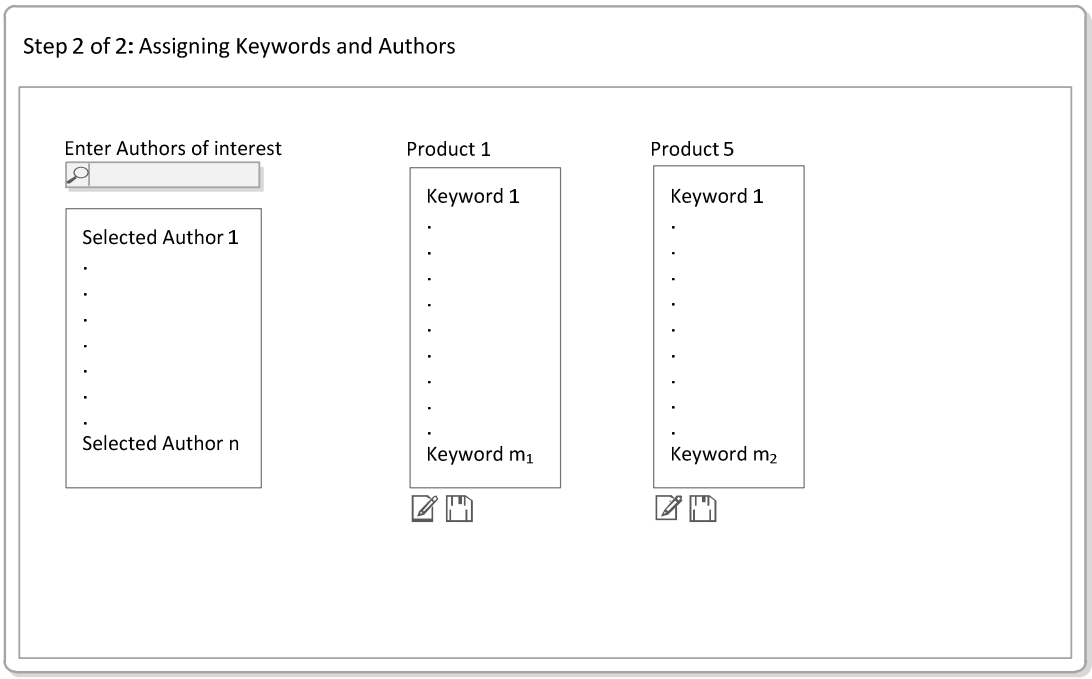

Figure 3 Expert-interface: Manage related authors and keywords for selected products

It is envisaged that once the user starts typing in the corresponding box to look for authors, suggestions of matching author-names from Twitter and some further context will be displayed. For each selected product, keywords are being pulled from DBpedia and populated in the list of related keywords. The user might add further keywords or remove existing ones.

3.9.2 End-User Interface

Figure 4 Wireframe for the end-user interface

The end-user interface allows the user to select a product of the previously created list by the expert. Then it shows relevant entries in four columns- one for each algorithm. The columns will not give an insight of which algorithm is used. They will be called A, B, C and D in order to allow an un-biased ranking. The number of entries will initially be set to 10 per column; the user-evaluation might help to find the best number of entries which allows enough detail and at the same time does not overload a user. The sentiment analysis is not visible for the end-user, it will be enabled or disabled through the instructor and results in two different setups that visually look the same, but present different data. The participants will be asked to rank each column and to specify which column presents the most relevant content.

3.10 Summary of System Design

An initial crawl of Twitter has been performed in order to gain an overview and a grasp of the topics that are being discussed actively. Apple products have been chosen as domain of interest due to its high popularity: they are actively discussed both by casual users as well as by news sites and technology blogs. Four algorithms were designed: A random selection of entries (A), the usage of social signals (B), an expert-enhanced list of authors, ranked randomly (C) and a combination in which the expert-enhanced authors are ranked by the

social signal (D). The number of followers was chosen as social signal because it provides a clear indicator for the popularity of a user and is easily accessible. The key concepts for two user-interfaces have been defined. Besides the evaluation of the performance of the four algorithms, a sentiment mode which is aiming to only select polarising content will be enabled and evaluated on top of all displayed content. In the next chapter, details about the implementation will be presented.

4 Implementation

4.1 Overview

In this chapter the implementation details of the designed system will be described. An overview of the classes and database structure to support the defining of semantics and to allow the storage of relevant Social Media data will be presented. Further, details on how external semantics, the Social Media data and the sentiment of this data were accessed through various interfaces will be discussed. The Semantic Enhancer, Twitter Crawler, Ranking Engine, Sentiment Analysis and the User Interface components will be presented. The process of decreasing the size of the collected data-set and the steps to optimise the performance will be illustrated. Finally, duplicate-detection and elimination as well as the development of the user interface will be discussed.

4.1.1 Classes and Database-Structure

4.1.2 Class Overview

To support the various operations needed, the following classes have been implemented:

- DBpedia (PHP): Querying DBpedia through SPARQL, built with the RDF API for PHP[27]
- MySQL (PHP): Wrapper class for querying the database
- ProductTweets (PHP): Selection and ranking of Tweets, duplicate recognition and spam detection
- TwitterDownloader (Java): To receive Tweets based on defined semantics, built with the Twitter4J library[28], textcat for language recognition[29] and JDBC drivers for a database connection[30]
- SentimentAnalyser (Java): To gather the sentiment score of relevant Tweets, built with the Alchemy API[31]

[27] http://www4.wiwiss.fu-berlin.de/bizer/rdfapi
[28] http://twitter4j.org
[29] http://textcat.sourceforge.net
[30] http://www.mysql.com/products/connector
[31] http://www.alchemyapi.com

4.1.3 Database Structure Overview

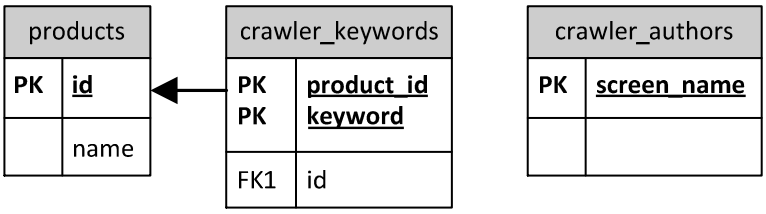

Figure 5 Database schema to store semantics

Figure 5 shows the schema for storing the semantics which will be used to query Twitter. Product names will be suggested based on the data in DBpedia and then the selected products by the expert will be stored with an automatically generated identifier. Keywords will be associated for each product. The authors are selected based on their screen name in Twitter.

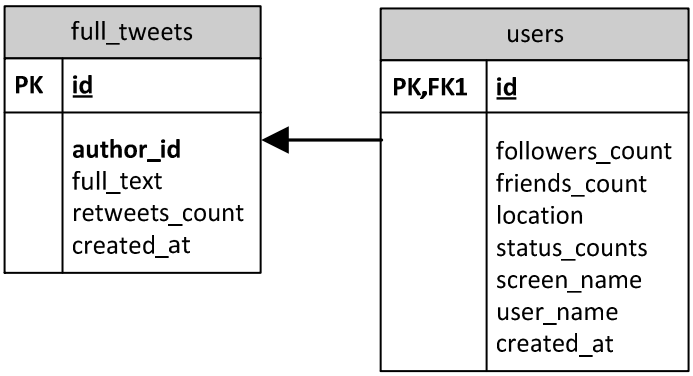

Figure 6 Database schema to store received data from Twitter

Figure 6 shows the two tables to store the data form the Social Network. Authors and Tweets will be stored separately because of the potential of multiple entries by the same author. For both the Tweet and the author a date is stored in the “created_at” field; for the author this is the date when the account has been created. Demographic data is stored for each user in order to support filtering and to allow accessing some background information if needed.

This is a minimalistic representation of the architecture restricted to only the tables that are needed for this design. Further tables for statistics existed, for instance to count the number of times a word occurred in all Tweets.

4.2 Semantic Enhancer

As presented in the design chapter, DBpedia will be utilised to gain knowledge about products and related keywords. These products and related keywords will be presented to an expert who can decide whether they should be used in order to crawl Twitter. DBpedia allows three different options of accessing the data (see 2.4.2). Since the amount of data required was relatively small it became clear that the querying with SPARQL was the most suitable option for this research.

In order to receive the products by Apple Inc. the following SPARQL-query is used:

```
SELECT ?productLabel ?product
WHERE
{
   {
   <http://dbpedia.org/resource/Apple_Inc.> dbo:product ?product.
   ?product rdfs:label ?productLabel.
   FILTER langMatches( lang(?productLabel), 'en').
   }
}");
```

Source Code 2 Receiving products by Apple Inc. from DBpedia with SPARQL

The resource Apple_Inc.[32] defines the page of interest in DBpedia, the tag dbo:product allows to query the products by this company. Further, the language is restricted to English results only.

[32] http://dbpedia.org/page/Apple_Inc.

In order to receive related keywords to a product, the following query is used:

```
SELECT ?productLabel ?productAbstract ?relatedSubjectLabel
WHERE
{
   {
   ?product rdfs:label \"".$productName."\"@en.
   ?product dbo:abstract ?productAbstract.
   ?product rdfs:label ?productLabel.
   ?product purl:subject ?relatedSubject.
   ?relatedSubject rdfs:label ?relatedSubjectLabel
   FILTER langMatches( lang(?productAbstract), 'en').
   FILTER langMatches( lang(?productLabel), 'en').
   FILTER langMatches( lang(?relatedSubjectLabel), 'en').
   }
}
");
```

Source Code 3 Receiving related keywords to a product from DBpedia with SPARQL

For this query, a product name is required in order to define which related keywords should be pulled. This product name must be equal to one of the results from the previous query (Source Code 2). The tag “dcterms:subject” is used in order to pull the related subjects and their label. Filters are used again in order to restrict all content to be in English language only.

4.3 Twitter Crawler

After the semantics have been received from DBpedia, the expert selected products of interest and used the suggested keywords to create his view on the domain (see Appendix A for the defined semantics).

The next step is to query Twitter according to these defined semantics. The crawler was developed by using Java and Twitter4j, a free, unofficial library for the Twitter API which supports the receiving of Tweets and user profiles in an object-oriented approach, OAuth (Twitters authorization system) and throttling to adapt to Twitters rate limitations[33].

[33] https://dev.twitter.com/docs/rate-limiting

The Java-based client is using the MySQL JDBC drivers to connect to the database which stores the authors and keywords of interest. This content is used as attributes for the Twitter filter-stream, which allows the following of particular users as well as defined keywords on Twitter.

```
public void onStatus(Status status) {

TextCategorizer guesser = new TextCategorizer();
String category = guesser.categorize(status.getText());
if(category.equals("english"))
{

      SimpleDateFormat df = new SimpleDateFormat("yyyy/MMM/dd
HH:mm:ss");
      Calendar currentDate = Calendar.getInstance();
      System.out.println("["+df.format(currentDate.getTime())+"]
"+status.getUser().getScreenName() + ": " +status.getText() );

      insertFullTweet(status);
      insertUser(status.getUser());
      insertWordOccurences(status);
      System.out.println();
  }

}
```

Source Code 4 Event of a Tweet qualifying based on the defined semantics

The method onStatus(Status status) is being called when a Tweet that qualifies is received. The status object contains all information available (e.g. the full text, date and the number of Retweets). It also incorporates the user object which holds the information about the author. First, Textcat is used to discard Tweets that are not in the English language. Textcat uses an N-Gram based classifier to categorize the text [21]. Further, the methods insertFullTweet(status) and insertUser(status.getUser) are called to store the the relevant information from the Tweet and the author in the database. The author will only be stored if it does not exist in the database yet. The method insertWordOccurrences(status) is used to count the number of occurrences of each word in the Tweet for statistical purposes (see 3.4).

4.3.1 Characteristics of Gathered Data

Within the time period from 2011-06-13 to 2011-06-28 (15 days) a total of 2,871,723 Tweets and 1,215,763 user profiles have been downloaded. The entries that were

composed by the expert-enhanced set of authors summed up to 6,919 (0.241% of the complete dataset).

The average length of a Tweet was 98.04 characters, when only looking at the expert-enhanced Tweets, the average length was 98.29.

The average number of followers per user was 696 (overall), 637 (non-experts), and 355,024 (expert-enhanced set of authors). The very high number of followers for the expert-defined list of authors was predictable, since most of these authors are well known technology blogs with high popularity. Some of the biggest sites included Mashable[34] (2,351,241 followers), TechCrunch[35] (1,706,776 followers) and Guardian Tech[36] (1,661,071 followers). However, the average number of followers overall was quiet high also, which can be explained by the fact that all of the listed users at least published one post on Twitter. Statistics show that most people with less than 10 followers never published any information on Twitter (see 2.2.4).

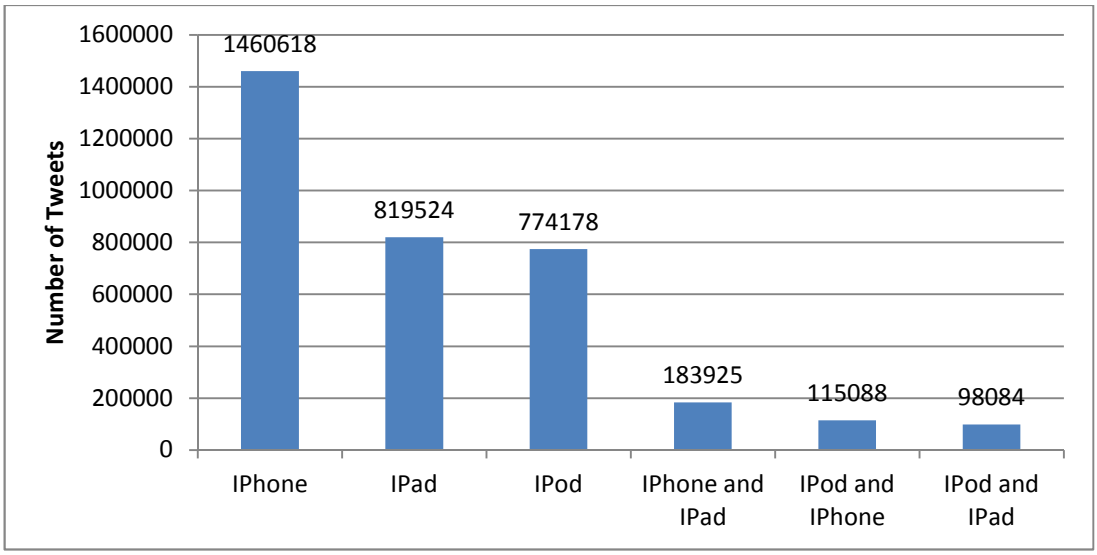

Figure 7 Most popular terms by occurrence in Tweets

Figure 7 compares the popularity of different terms and correlations of terms. IPhone was the most popular word which occurred 1,460,618 times, followed by IPad and IPod. Correlated, the combination of IPhone and IPad occurred the most, probably because of the use of the same operating system and applications running on both devices.

[34] http://mashable.com
[35] http://www.techcrunch.com
[36] http://www.guardian.co.uk/technology

Expert-enhanced Crawl	*Initial Random Crawl*
The	Have
Iphone	I'm
My	Not
L	…
To	Love
On	Lol
A	Don't
RT	How
And	Know
For	Good
Ipad	Got
Ipod	Now

Table 1 Most common words in Tweets, Selected Dataset vs. Random Selection

Table 1 shows the most popular words in the selected dataset and in the initial crawl (see 3.4). Applying the semantics led to the effect of listing IPhone, IPad and IPod in the top 12 words ranked by number of occurrences. Related terms such as "app" or "Apple" were also ranked much higher in the expert-enhanced dataset.

4.4 Building the Data Corpus

The data crawler was set up to run for 15 days and collected roughly 2.8 million Tweets. The next step was to query the data and to look for a product name and at least one related keyword. The Boolean Full-text Search feature of MySQL[37] was used to perform this task: The full text of each Tweet was matched against the product name and at least one additional keyword. In this section, the performance issues and the creation of a subset of the data to overcome this problem will be discussed.

4.4.1 Motivation for Building a Subset of the Social Media Data

The queries to receive Tweets of interest were built dynamically by selecting the keywords and a product name from the database.

[37] http://dev.mysql.com/doc/refman/5.1/en/fulltext-boolean.html

```
SELECT
   t.full_text, u.screen_name
FROM
   full_tweets t, users u
WHERE
   MATCH(full_text) AGAINST('+IPad +("Tablet" "ITunes" "iOS" "Wi-
   Fi" "Touchscreen" "Portable" "Multi-touch" "Multitouch"
   "Appstore" "Icloud" "Iphone" "FaceTime" "Smart Cover" "HD
   Video" "Microphone" "mp3" "8gb" "16gb" "32gb" ))' IN BOOLEAN
   MODE)
   AND u.id=t.author_id
ORDER BY RAND()
LIMIT 10
```

Source Code 5 Typical SQL query to select a set of Tweets

Source Code 5 shows a typical query which would return the relevant entries for the product "IPad" (simplified version, not including any spam detection, duplicate elimination or expert-enhanced authors). To enable the database to recognise short words such as "mp3", the full text minimal string length has been set to three (ft_min_word_len=3). The MySQL server has been optimised to handle large datasets through increasing associated memory, creating indices and using the MyISAM storage engine[38] rather than InnoDB[39] because it is faster for applications with a lot of heavy reading queries. However, the performance still was not sufficient, the full text search on the dataset of 2.8 million Tweets took about 400 seconds for an individual query. It was desired to run multiple queries and to perform further operations on the results which would lead to very long computations times.

Two possible approaches to increase the performance were evaluated:

1. Create a subset of relevant data in order to decrease the number of rows which need to be matched against the search string
2. Implement a custom caching architecture to store the relation between products and related Tweets

[38] http://dev.mysql.com/doc/refman/5.0/en/myisam-storage-engine.html
[39] http://dev.mysql.com/doc/refman/5.0/en/innodb-storage-engine.html

The second option would ensure that the complete dataset is used and, for instance, a random select would have the possibility to select from all collected data. However, this approach is non-trivial to implement, whenever the list of keywords changes, the entire cache needs to be updated to add or remove Tweets that qualify. Further, it was necessary to collect a large amount of data in order to get enough matching entries, but it seemed very feasible to select this relevant data and to reduce the amount of data to a much smaller set which is actually needed for this research. For instance, displaying the top 10 Tweets by the number of followers will always return the same results. It is not necessary to store more data than these 10 entries; therefore a lot of unnecessary data existed in the dataset. Another reason why it was desired to reduce the size of the dataset is caused by the request limitations for third party providers for sentiment analysis (details on these limitations will be discussed in section 4.5). In the next section, the process of reducing the size of the dataset will be presented.

4.4.2 Methodology to Generate Subset of Data Corpus

The goal of the creation of the subset was to increase the performance significantly while maintaining all the data that is required for this research. Section 3.7.3 described the different algorithms that will be evaluated. To support the expert-enhanced algorithms which use a defined list of authors (see 3.7.3.3 and 3.7.3.4), all Tweets that are created by one of the defined authors were copied in the subset of the data corpus. The selected products were “IPhone”, “IPod” and “IPad”.

The next step is to support the random selection (see 3.7.3.1) and the ranking by social status (see 3.7.3.2). To achieve this, for each of the three products 5,000 random Tweets and the first 1,000 Tweets ordered descending by the number of followers were selected. These numbers ensure that there is enough data for randomly selecting entries and for selecting the most popular entries by social status. For experimental reasons, the top 1,000 entries ordered by number of Retweets also were copied in the subset. To verify the dataset, queries have been run and compared on both the subset and the initial dataset.

Overall, 18,266 Tweets and 13,939 individual users were selected in the process of reducing the size of the original dataset. The average length of a Tweet was 108, which is slightly higher compared to 98 from the original dataset. The average number of followers of the 13,939 users is 6,501, which is significantly higher compared to the 696 of the original dataset. This is caused by the high reduction of “regular users”; as presented in 4.3.1, the

average number of followers for experts was 355,024. All the expert-users have been included in the subset of the data; therefore the average number of followers has been raised significantly.

4.5 Sentiment Analysis

In the design chapter the approach for the sentiment analysis was discussed and it was envisaged to use a third party provider to receive the sentiment score of the Social Media data (see 3.8). After an analysis of possible technologies and third-party providers, the Alchemy API[40] was chosen to compute the sentiment of the dataset. By default Alchemy allows 1,000 requests per day; the dataset of 18,266 would have taken 19 days to analyse. Alchemy raised the limit to 30,000 requests per day for this academic research; therefore the sentiment analysis could be performed quickly. The full data-set of 2.8 million Tweets would have taken about three months to enrich with sentiment scores based on the Alchemy API, this should be considered when developing a large scale system.

To support the storing of the sentiment for each Tweet, the database model was extended with two additional columns: “sentiment_score” and “sentiment_type”. The provided score ranges from -1 (very negative) to +1 (very positive). The sentiment_type simplifies the result in form of a string with a lesser granularity (negative, neutral or positive). For this work, a java-based crawler to collect the sentiment was developed, similar to the crawler for Twitter. Entries were selected from the database and for each row a request to the Alchemy API was sent. The returned sentiment score and type was saved in the database. The default response format is JSON[41]; by utilising the official Alchemy-java class[42] the API could be accessed directly, the results were returned as Java Objects of the type “Document” and could easily be traversed. The only problem that occurred was that a small set of 138 entries was not recognised as English language by the Alchemy API and therefore the sentiment could not be calculated.

[40] http://www.alchemyapi.com/api/sentiment
[41] http://www.json.org
[42] http://www.alchemyapi.com/tools

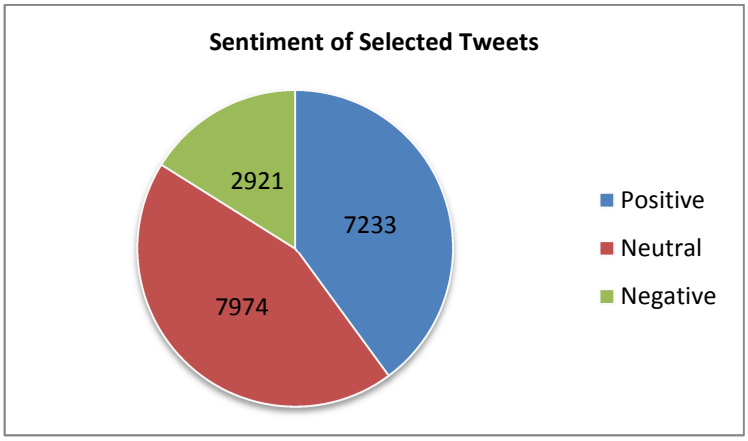

Figure 8 Sentiment Statistics of the Selected Data Corpus

The average sentiment score on the data was 0.031, indicating a slight tendency to positive entries. Most entries were neutral, summing up to 7,974, followed by positive entries with a count of 7,233. Only 2,921 Tweets were categorised as negative.

4.6 Ranking Engine

Four algorithms to order the qualified Tweets have been designed in section 3.7.3: A random selection (A), selection ranked by social status of the author (B), random selection of entries by an expert-enhanced set of trusted authors (C) and entries by an expert-enhanced set of trusted authors, ranked by their social status (D).

The data was selected by utilising the Boolean Full-text Search feature of MySQL, as described in 4.4. Additionally, the creation date of the account and the creation date of the Tweet were compared. The Tweet was selected only if the difference was higher than 2 days. The goal of this process was to eliminate Spam, similar as presented in related work [6].

4.6.1 Detection and Elimination of Duplicate and Partial-Duplicate Entries

At first, it seems like the algorithms do not correlate, but it became apparent that because the results were displayed at the same page, it is important to detect duplicates and partial duplicates in order avoid showing the same data in different columns. The algorithms C and D naturally do contain different content than A and B, since they only show expert-enhanced content while A and B only show non-expert enhanced content. However, there still needs to be a solid detection of partial duplicates, because of the special characteristics

of Twitter. Identical entries could easily be removed by using the distinct optimization function of MySQL[43].

However, in several cases entries were Retweeted and extended, for instance with more hash tags or with a different external link, but basically showed the same content.

For instance:

> Tweet 1: 'Orbit' Exposé for iPhone Updated to Support iOS 4 http://ow.ly/5p3RA
>
> Tweet 2: 'Orbit' Exposé for iPhone Updated to Support iOS 4 http://j.mp/jXSKeQ

The included information is the same in both entries, only the external link changed and one of the two entries should not be displayed. For this work it was desired to select only one of the two entries, it was not considered which entry will be removed.

To achieve this, the Levenshtein distance between each qualified Tweet and all the previously selected qualified Tweets is computed with the PHP implementation of the algorithm[44]. The distance needs to be higher than the threshold which was set to 15 (at least 15 operations are needed to convert one string into another). This led to a successful elimination of partial duplicates.

A MySQL implementation of the Levenshtein distance algorithm[45] was also evaluated but had to be omitted because it reduced the performance significantly. Just one comparison with a string length of 78 characters led to query execution times that were roughly 4 times slower, which is not suitable for n^2 calculations. To support this in MySQL efficiently, a different caching technique would be required.

Another evaluated approach to detect partial duplicates was to utilise the Soundex string function of MySQL[46]. This method returns a string-representation of the sound of a string. Very similar sounding strings return the same value for the Soundex. However, after testing it became clear that this approach would eliminate some of the partial duplicates but it was not suitable for detecting higher differences in content. The sound was not a clear indicator for the similarity in Tweets.

---

[43] http://dev.mysql.com/doc/refman/5.0/en/distinct-optimization.html
[44] http://php.net/manual/en/function.levenshtein.php
[45] http://www.artfulsoftware.com/infotree/queries.php#552
[46] http://dev.mysql.com/doc/refman/5.0/en/string-functions.html#function_soundex

4.6.2 Algorithm Execution

The algorithms are executed sequentially from where the least data is available to most where the most is data available (D, C, B, A). One issue was the estimation of the needed extra entries in order to support the duplicate detection. The detection was performed with PHP, therefore a dataset bigger than needed had to be selected by MySQL, prior to the duplicate elimination. For instance, if algorithm D displays 8 Tweets and algorithm C randomly picks an entry that is a partial duplicate to one of the 8 entries of D, a total of 9 entries have to be pulled from the database to display 8 entries in algorithm C. This could be achieved through an ad-hoc pulling whenever more data is needed. However, since the queries were complex and slow to execute, the used approach was to always pull at least twice the amount of data in order to have a big enough set of Tweets to work with.

The partial duplicate detection was not only comparing the currently selected entry to all previously displayed entries of other algorithms but also to previously displayed entries from the same algorithm. This approach allows to filter the MySQL results and to only print unique entries.

The MySQL queries were built dynamically; to support the sentiment analysis a Boolean variable was used. If the mode was turned on, the query was extended with a “where clause” to select only negative or positive entries.

4.7 User Interface

The key concepts of the user interface were described in 3.9: Two different frontends to support the end-user exploration of Social Media and the expert enhanced annotation of a domain were developed. In both cases a web-based solution was implemented based on state of the art technologies, such as HTML/CSS layouts combined with PHP and Ajax to allow server-site operations and dynamic updates of the interfaces.

4.7.1 End-User Interface

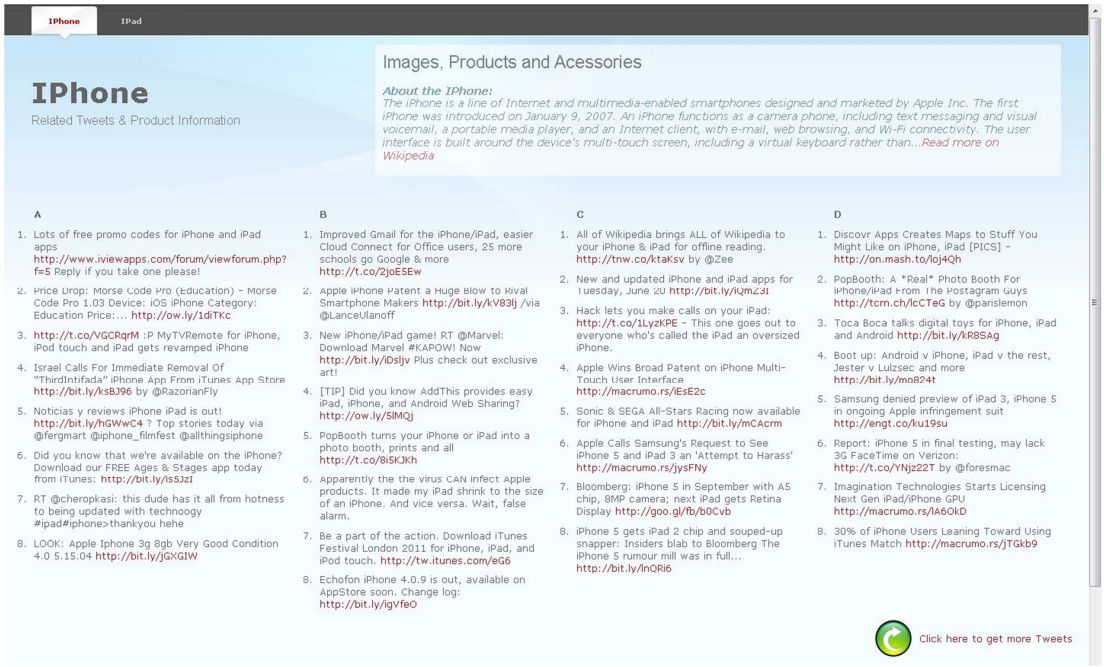

Figure 9 End-User Interface

In this work, the term End-User refers to the person evaluating the Social Media content selected by the different algorithms. The four columns were introduced for the reason of comparing and ranking selected data, in a final product these columns would most likely be merged to one column which intelligently presents Social Media content based on defined semantics and algorithms.

This End-User interface allows a user to choose between the available products (IPhone and IPad). In the main content area, the selections of the algorithms are displayed in the columns A, B, C and D. To improve the exploration, the abstract of each product has been pulled from Wikipedia and displayed on the top of the page. External links have been automatically transformed to clickable HTML links. A large refresh button on the bottom right side of the panel allows the receiving of more content.

4.7.2 Expert Interface

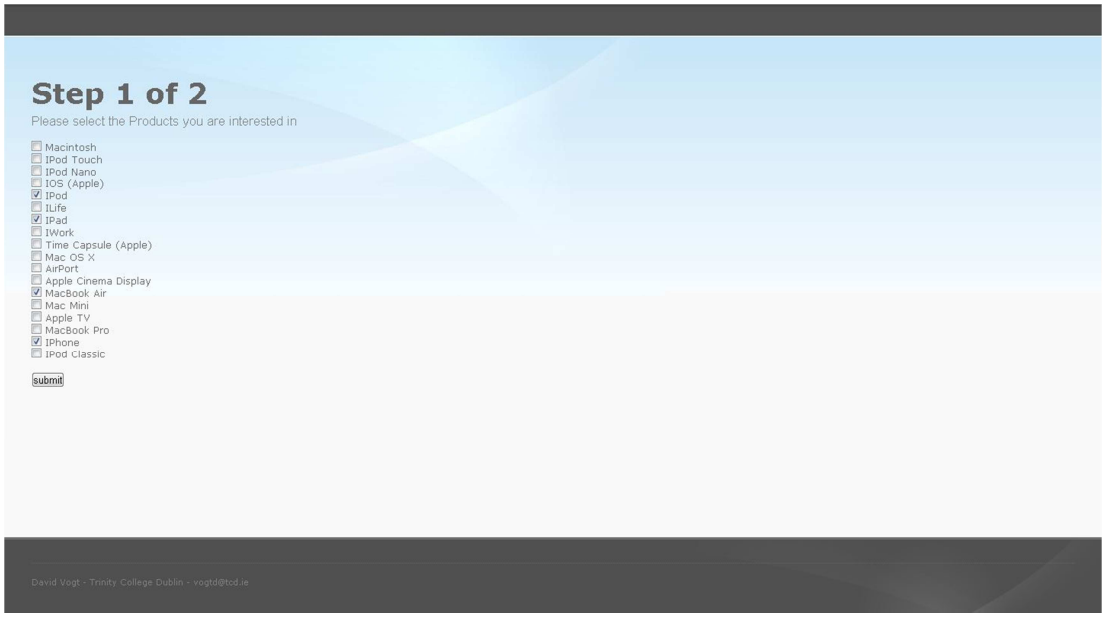

Figure 10 Expert-Interface, Step 1

The purpuse of the expert-interface is to support the process of defining semantics which will be used to download data from Twitter and to rank the selected data. When the expert-interface is called for the first time, a selection of available products is presented and the user can select the products of interest. This list is pulled from DBpedia by the Semantic Enhancer component.

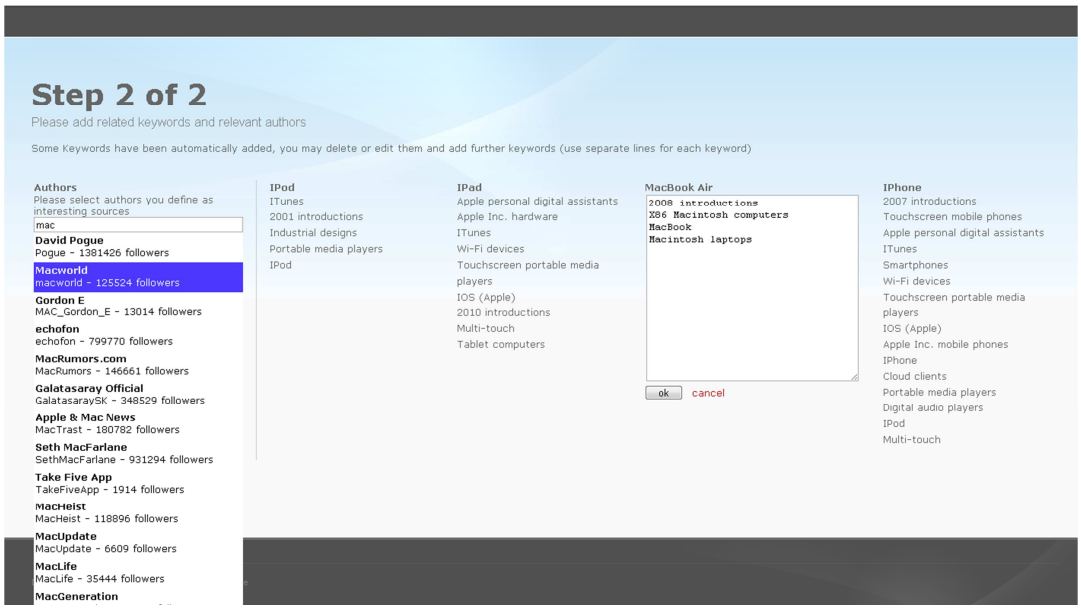

Figure 11 Expert Interface, Step 2

In the second step, the selected products are listed horizontally and the very first column allows adding the authors who are considered trusted experts of the domain. As soon as the user starts typing in the input form, suggestions of Twitter usernames will be displayed. This is implemented by utilising the search functionality of Twitters API[47]. The effect of displaying and dynamically updating the list of proposed authors is created by using Script.aculo.us[48], a cross-browser JavaScript library to build effects and Ajax operations.

For each product, a list of related terms is pulled from DBpedia and populated in the editable area underneath the product name. The user can add, edit and delete keywords for each product and save these changes through an Ajax request.

4.8 Summary of Implementation

In this chapter, the implementation of the key components was presented. Two java-based crawlers to receive Social Media content and to compute the sentiment of the selected data have been presented. The process of accessing DBpedia through its SPARQL interface and how the gathered data is used to semantically enhance a knowledge domain and to make suggestions of relevant keywords to an expert user were discussed. Further, the challenge of handling the large dataset of 2.8 million Tweets and how this set was reduced to a much smaller size have was illustrated. Four different algorithms to allow the selecting of relevant data based on different signals were implemented. One of the key challenges was the detection and elimination of semi-duplicate entries, which was addressed by using the Levenshtein distance algorithm. Two user interfaces to annotate a knowledge domain and to display tailored views on the Social Media content were developed. In the next chapter, the implemented user interfaces and the performance of the different algorithms will be evaluated.

[47] https://dev.twitter.com/docs/api/1/get/search
[48] http://script.aculo.us

5 Evaluation

5.1 Overview

The goal of the evaluation is to analyse the degree of usability of the Peregrine system and to verify the assumptions of the design section (3.5). To achieve this, two different experiments will be performed: First, an end-user interface which selects and displays entries according to different signals will be evaluated. Secondly, an expert-system to create these tailored user-interfaces and to manipulate the algorithms to select data from Social Media will be tested. In this section, the approach for this survey, results, observations and an analysis will be presented. The questionnaire and the goals of the assessment were part of the design process of this work and have been built to a high degree of detail before the start of the implementation.

5.2 Approach and Goals of Assessment

5.2.1 Goals

The key aims for the evaluation lay in two areas: usability and content relevance. To assess usability, Thomas Tullis and William Albert have identified key questions that will be used as a general guideline for this survey [22]:

- What are the most significant usability issues that are preventing users from completing their goals or that are resulting in inefficiencies?
- What aspects of the product work well for the users? What do they find frustrating?
- What are the most common errors or mistakes users are making?
- Are improvements being made from one design iteration to the next?
- What usability issues can you expect to remain after the product is launched?

Besides usability tests, the performance of the algorithms and the relevance of selected Tweets will be analysed. To achieve this, the following questions will be addressed:

- What are the strongest signals for relevance when selecting entries from Twitter?
- Does the sentiment of the entry have an impact on the perceived quality?
- How relevant and useful are these entries generally?
- How much user-control over the content is desired?

- Would users look at expert-enhanced mediated Social Media content to gain more knowledge about products?
- What are the needed characteristics of a system to create expert-views on Social Media content?

5.2.2 SUS

Different approaches to measure usability exist; one widely used framework is SUS – System Usability Score [23]. This is a very simple but powerful questionnaire which consists of only 10 questions. Despite its simplicity, it often proved to be very reliable in comparison with other approaches [24]. Each question of the SUS-Framework addresses an overall opinion on the usability (e.g. “I found the system unnecessarily complex”) and can be rated from 1 (strongly disagree) to 5 (strongly agree). The SUS framework has been used twice in this experiment: For the end-user interface, which is a prototype to displays selected content from Social Media. And secondly, for the expert-interface, which allows the annotation of semantics to describe a knowledge domain. The complete list of the questions for SUS and algorithm performance will be presented in section 5.3.

5.2.3 Evaluation Plan

Two separate experiments to evaluate the two systems will be performed: An end-user evaluation to measure the impact of different signals and an expert-interface evaluation to measure the usability of the annotation tool. For the first experiment, an external expert will define the semantics which will be used to generate the displayed content. In the second experiment, only the process of defining the semantics will be evaluated. Hence, the Social Media selected based on the defined semantics by the end-users will not be evaluated.

Firstly, the participants will be asked background questions, such as their familiarity with Social Networks and their discipline of studies. Then, the end-user interface will be shown which consists of two products (“IPhone” and “IPad”).

Four columns with related content from Social Media projects will be displayed for each product:

Column name	*Algorithm used*
A	Random (random selection of entries, see section 3.7.3.1)
B	Social Status (entries ranked by number of followers, see section 3.7.3.2)
C	Expert-Enhanced (entries must be written by a defined set of trusted authors, sorted randomly, see section 3.7.3.3)
D	Combined (entries must be written by a defined set of trusted authors, ranked by number of followers, see section 3.7.3.4)

Table 2 Column identifiers and corresponding algorithms for the end-user evaluation

The user has no insight about which algorithm is used for each column. He will be asked to rank the algorithms for two different setups in order to support the envisaged sentiment analysis. In the first setup, all entries were shown according to the algorithms with no restriction. In the second setup, only entries that were considered as either positive or negative will be shown. The goal is to evaluate if the sentiment has an impact on the perceived relevance of the selected data.

For the evaluation, the ranking of the algorithms will be calculated as a simple linear score:

- First position equals 3 points
- Second position equals 2 points
- Third position equals 1 point
- Fourth position equals 0 points

For instance, the ranking D,B,C,A would result in D gaining three points, B two points, C one point and A zero points. Therefore, the algorithm with the highest sum over all participants' individual scores is considered to be the best one overall. The participants will also be asked to state if they noticed an increase or decrease in one of the two setups. This allows to observe if the sentiment mode improves the quality overall. Once the participants have used the Peregrine system and ranked the algorithms, further questions on the basic usability by using the SUS scores will be asked.

After the evaluation of the end-user interface, the expert-interface will be tested. This interface allows users to select products that they are interested in and to relate keywords

that they associate with these products. Both the products and the keyword suggestions will be pulled from DBpedia. Further, the user can add authors that he considers as important sources for these products. This can be achieved through filling in a form which automatically suggests users from Twitter based on the input. The evaluation consists only of a SUS-Score, mainly due to technical limitations (it would take a long time to gather the data based on the expert-selections).

5.2.4 Participants

For the experiments, 11 participants were chosen on a voluntary basis. All of the participants were postgraduate students at Trinity College Dublin from different disciplines such as computer science, law and natural sciences.

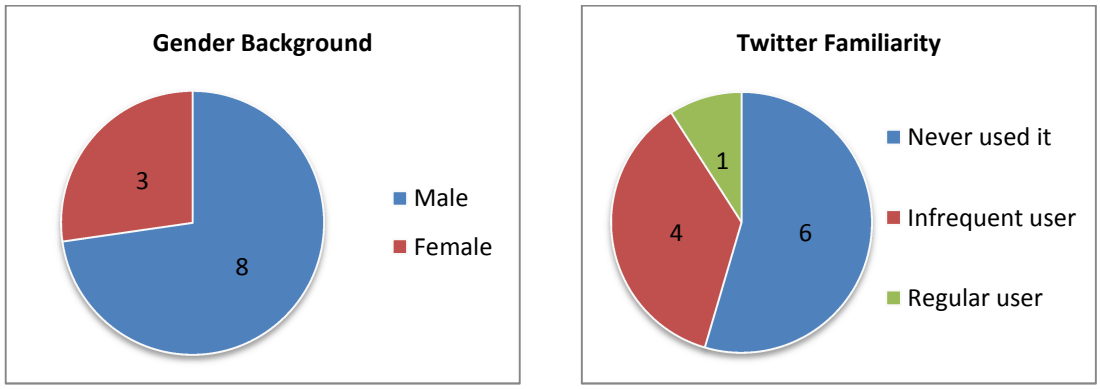

Figure 12 Participants background

Participants will generally be referred to as "he" because the majority was male, even though a number of women participated in the experiment. The majority of the users stated to never have used Twitter, the second most common experience level stated was "infrequent user". As a result, very little knowledge of Twitter and its nature of communication and publishing content were expected. All users were asked to perform the same tests and to answer the same questions. The results of the first two user evaluations were used as a pilot-test in order to find initial flaws and problems in the Peregrine system. Hence the data of 9 participants was used for the main experiment.

5.2.5 Instructing Participants and Planning of the Evaluation-Sessions

Each session was planned to last for no more than 45 minutes. In the beginning, the participants were informed verbally about the product and the areas that will be assessed. They were told that the performance of different algorithms will be evaluated, though no information about any signals or characteristics of the algorithms was given. Also, no

disclosure about the difference of the two setups (sentiment on or off) was given. Since the evaluation was aiming to gather information about the data quality, rather than the ease of performing certain tasks with Peregrine, there was no focus on time-efficiency and the total time of each session will not be assessed in this work.

Most questions were easy to understand and did not require explanation. The only part which needed to be discussed in detail was about the ranking of algorithms by the perceived relevance. The general definition of relevance in this work is that "an entry is relevant when it improves the knowledge about the product without the need to have further background knowledge" (3.7.1). This is an intentionally vague definition because users might have their own ideas of what is useful and therefore relevance can be highly biased. However, in the end of this chapter it will be discussed how this vague definition led to some issues in the evaluation.

The following list was given to participants in order to make it easier to classify entries. An entry may be relevant when it helps to:

- Discover opinions about the product
- Discover positive/negative experiences with the product
- Discover alternatives/competitors of the product
- Get special deals to buy the product or related products
- Find out about functionality and applications of the product
- Find links to other resources of interest related to the product
- Discover news and developments about the product

5.3 Questionnaires and Metrics

The questionnaire consisted of two parts: The end-user interface and the expert-interface. The end-user interface was evaluated first, and then the user was told that the expert-interface allows them to create their own semantics to find relevant Social Media content. While the end-user interface evaluation included in-depth questions about the quality of the data presented, the expert-interface evaluation just consisted of general usability questions.

5.3.1 End-User Interface

5.3.1.1 Overall Satisfaction with the Product (End-User SUS – Scores)

Q1	I think that I would like to use this system frequently
Q2	I found the system unnecessarily complex
Q3	I thought the system was easy to use
Q4	I think that I would need the support of a technical person to be able to use this system
Q5	I found the various functions in this system were well integrated
Q6	I thought there was too much inconsistency in this system
Q7	I would imagine that most people would learn to use this system very quickly
Q8	I found the system very cumbersome to use
Q9	I felt very confident using the system
Q10	I needed to learn a lot of things before I could get going with this system
Scale: 1 (Strongly disagree) to 5 (Strongly agree)	

Table 3 Questionnaire on overall satisfaction with end-user interface

5.3.1.2 Questions Regarding the Background of the Participant

	Question	Possible answers
Q11	How would you describe your familiarity with Twitter/Microblogging technologies	Expert Regular User Infrequent User Never used it Unfamiliar Term
Q12	The most recent degree that you have completed (or equivalent)	B.Sc. B.A. M.A. M.Sc. PhD Other
Q13	My gender is	Male Female

Table 4 Questionnaire on background information of the participants

5.3.1.3 Questions on the Algorithms and the Relevance of Selected Entries

	Question	Possible answers
Q14 (a,b)[49]	Please rate the Social-Network columns ordered by usefulness (for example: B | A | D | C would indicate that column B is the most useful and column C the least useful)	Any possible order of the columns A,B,C,D Most useful to least useful
Q15	Which setup provided more useful information?	First Setup Second Setup No Difference

Table 5 Questionnaire on relevance of selected entries

5.3.1.4 Questions on the Overall Satisfaction with the Selected Entries

Q16	The information from Twitter is useful and improves my knowledge about the product
Q17	The displayed entries were relevant to the product
Q18	I would like to have more control over what is being displayed
Q19	I would use this system when considering the purchase of electronic devices
Scale: 1 (Strongly disagree) to 5 (Strongly agree)	

Table 6 Questionnaire on overall satisfaction with selected content

5.3.2 Expert Interface

5.3.2.1 Overall Satisfaction with the User-Interface (Expert SUS-Scores)

Q E1	I think that I would like to use this system frequently
Q E2	I found the system unnecessarily complex
Q E3	I thought the system was easy to use
Q E4	I think that I would need the support of a technical person to be able to use this system

[49] This question was split off in a and b in order to support the ranking with two sentiment modes, where a) showed entries regardless of their sentiment and b) only showed entries with positive and negative sentiment

Q E5	I found the various functions in this system were well integrated
Q E6	I thought there was too much inconsistency in this system
Q E7	I would imagine that most people would learn to use this system very quickly
Q E8	I found the system very cumbersome to use
Q E9	I felt very confident using the system
Q E10	I needed to learn a lot of things before I could get going with this system
Scale: 1 (Strongly disagree) to 5 (Strongly agree)	

Table 7 Questionnaire on overall satisfaction of expert-user-interface

5.4 Pilot-Test

The goal of the pilot test was to get initial feedback on the Peregrine system and to find out if there are any major problems that a user would face. Also, it was important to identify if the tasks are too complex or if there are any flaws in the design of the questionnaire. The evaluation was performed with two participants and the results are presented in the following sections.

5.4.1 End-user System

5.4.1.1 Sus Scores

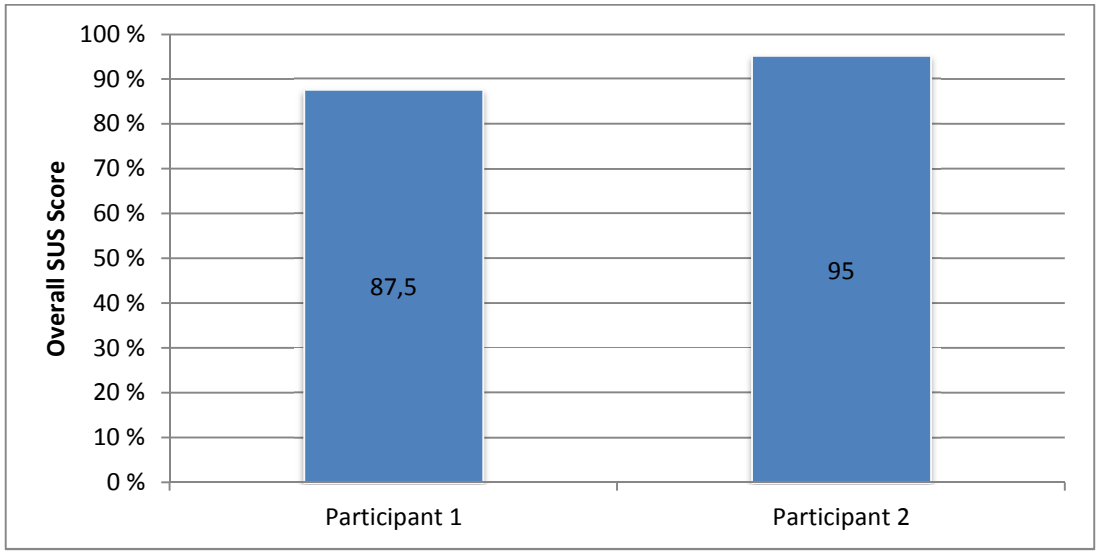

Figure 13 SUS scores for end-user interface, pilot test

The over-all SUS score average of the end-user interface was 91.25% which is indicating that the users felt very comfortable when using the Peregrine system. Figure 14 also shows

the individual scores as a spider diagram which illustrated that the participants agree on most questions.

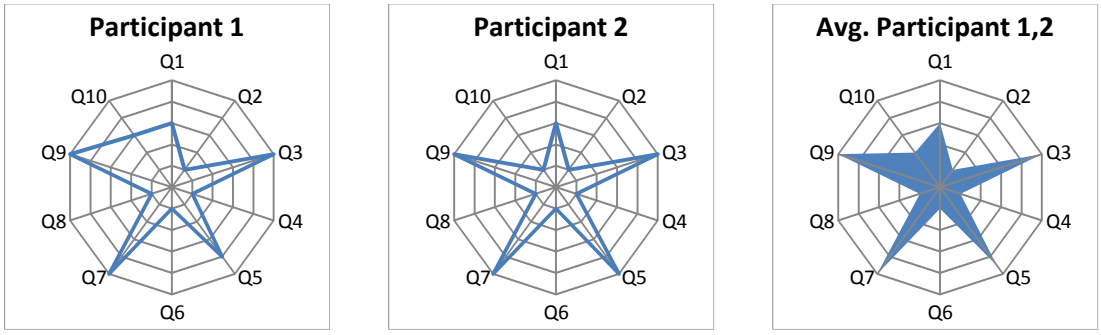

Figure 14 Individual SUS scores and SUS averages for the end-user interface, pilot-test

5.4.1.2 Algorithms Performance

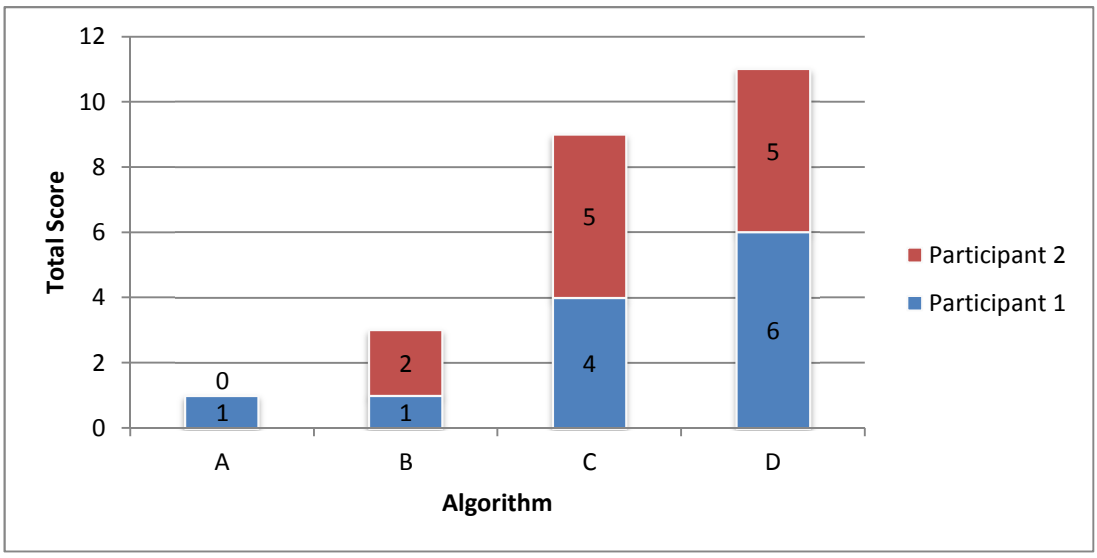

Figure 15 Algorithms ranking, pilot-test

Figure 15 shows the overall scores of the algorithms, separating the score of each individual user by colour. These initial results already support the hypothesis of this research that the expert-enhanced algorithms improve the perceived relevance of selected data. Column A, the random data, was ranked very poorly and only rated as number three out of four in one individual user-rating. Both test-users concurred that columns C and D (both expert enhanced) provide the highest quality.

5.4.1.3 *Overall Usefulness and Satisfaction with Selected Entries*

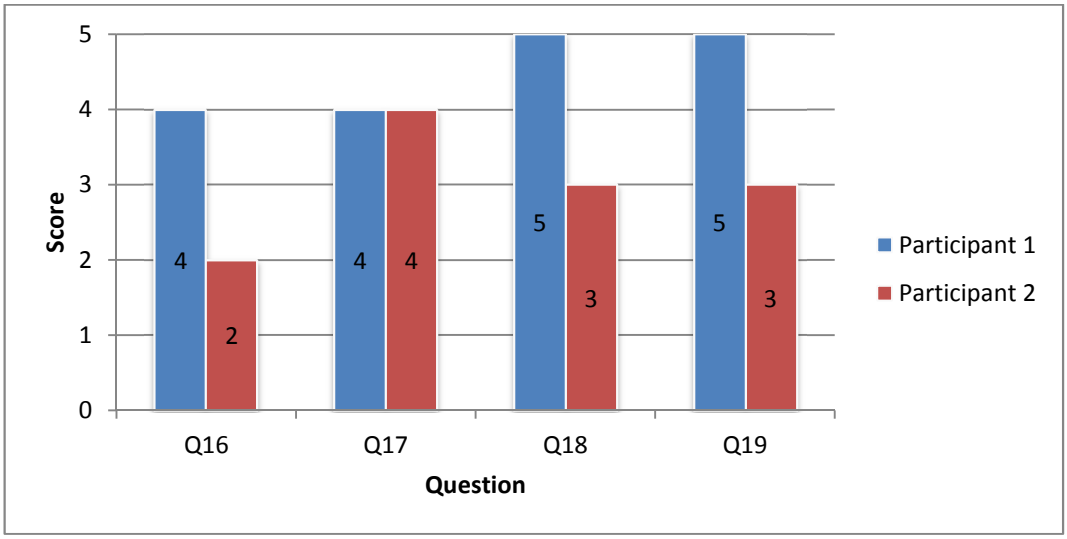

Figure 16 Relevance of Entries and User-Satisfaction Overall, Pilot-Test

The participants both agreed that the information displayed was relevant to the selected products (Q17). The rating on the other questions varied and with only two participants it is difficult to make any assumptions and analyses on these questions, this part will be of more value when working with more participants.

5.4.2 Expert System

5.4.2.1 *SUS Scores*

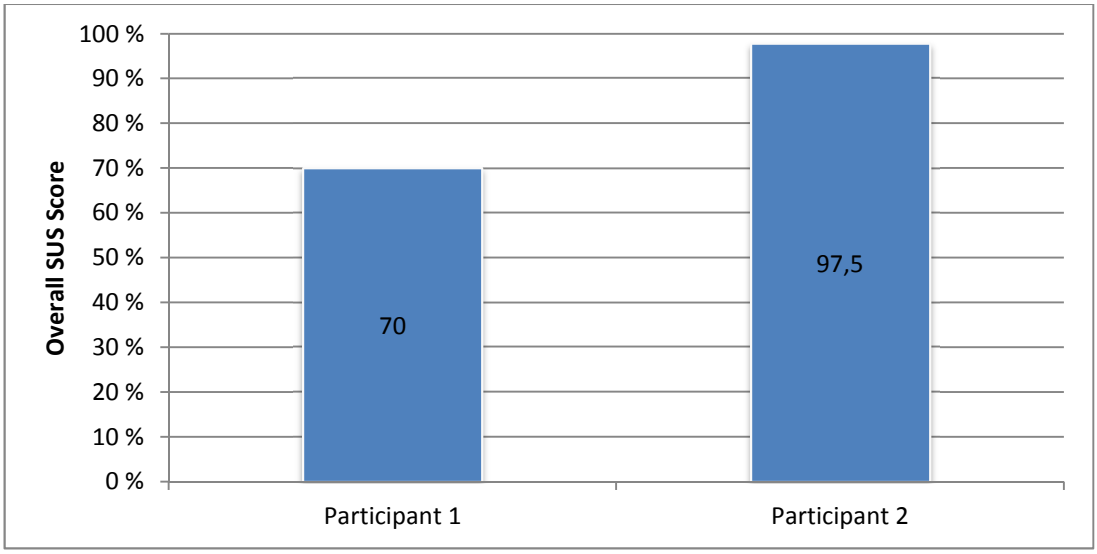

Figure 17 Overall SUS Score of the Expert System, Pilot Test

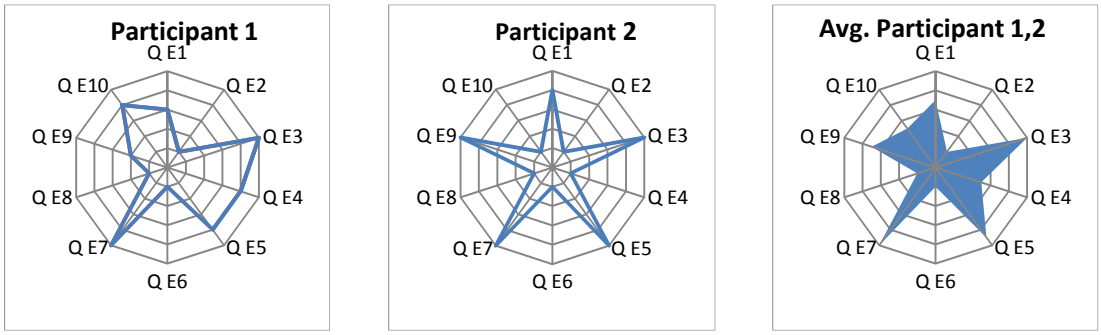

Figure 18 Individual SUS-Scores and SUS-Averages for the expert-interface, pilot-test

Surprisingly, even with coming from a non-technical background both participants found the expert-system easy to use. The SUS score average was 83.75%, participant two even rated the usability with 97.5%.

5.4.3 Observations and Findings of the Pilot-Test

It became clear that the amount of information presented to the user exceeded their capabilities of absorbing it. The experiments took longer than anticipated (over one hour instead of the desired 30-45 minutes). There seemed to be too much information to process, which also has been expressed by the participants in comments such as:

> "I don't like too much text on one website, I would prefer to get just 5-10 entries with options to check further on the next ten results" – Participant 2

The entries were presented as a simple list, separated with a horizontal line in between each row. Both people noted that the presentation is cumbersome sometimes and that they would like to have more structure, e.g. by simply using bullet points for each entry.

It could be observed that the participants both did not like to refresh the page in order to receive more entries, even after they have been told that this is a possible feature. This most likely is due to the amount of information to process. Another minor note by a user was that the need to scroll down to see the complete list of entries is a disadvantage.

In order to rank the algorithms, both users took notes in varying degrees of detail: One participant noted the overall relevance for each column (e.g. 60%, 30%) and then slowly came up with his overall ranking. The other participant rated each Tweet individually (helpful, neutral, not helpful) and then calculated the score based on these ratings.

5.4.4 Modifications of the System after the Pilot-Test

One very important insight was that it is very hard to process the amount of data offered by the Peregrine system. Two different setups, three products with four columns and 10 entries per column resulted in 60 entries to read and rate. To properly analyse Peregrine, it is necessary to refresh the page at least once, since some algorithms used random orderings and deliver varying qualities. As a result, when just refreshing every page once, 120 entries would have to be processed. To avoid this, the first adaption was to remove one of the three products. The product "IPod" seemed suitable to remove, because it generally was not covered very much in the data-set. Another step taken was to reduce the number of entries per algorithm from ten to eight.

To adapt the visualisation to the users' demands, the horizontal lines were removed and a simple numbering for each entry was placed instead. The number of entries per column was reduced from 10 to eight, which led to a decrease of the height of the page. To further address the problem of a need to scroll between entries, the user interface was launched in full-screen mode.

Further, to animate the participants to refresh the page more often, a big button with the label "Click here to get more Tweets" was introduced.

5.5 First Usability Test: End-User Interface

5.5.1 SUS Scores

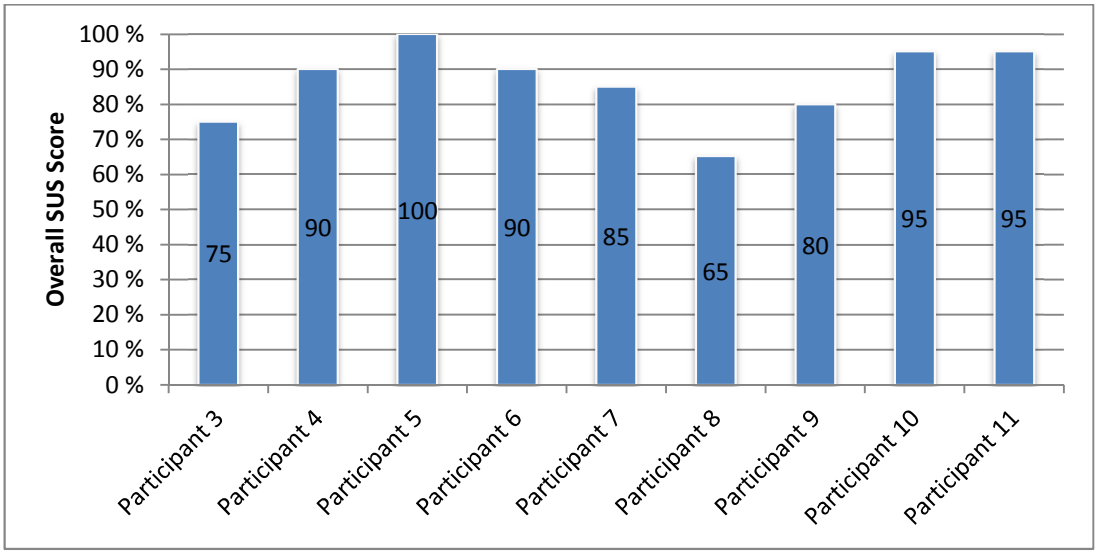

Figure 19 SUS-Averages for the end-user interface in the main-study

The average of the SUS-Score overall was 86.11%; Figure 19 shows the individual SUS-Scores for each participant. The standard deviation for these records was 10.48%, illustrating the high agreement on the usability. Figure 19 also shows the individual results in form of a spider diagram. The average star has a very even distribution which is proofing that most participants had the same tendency for all questions.

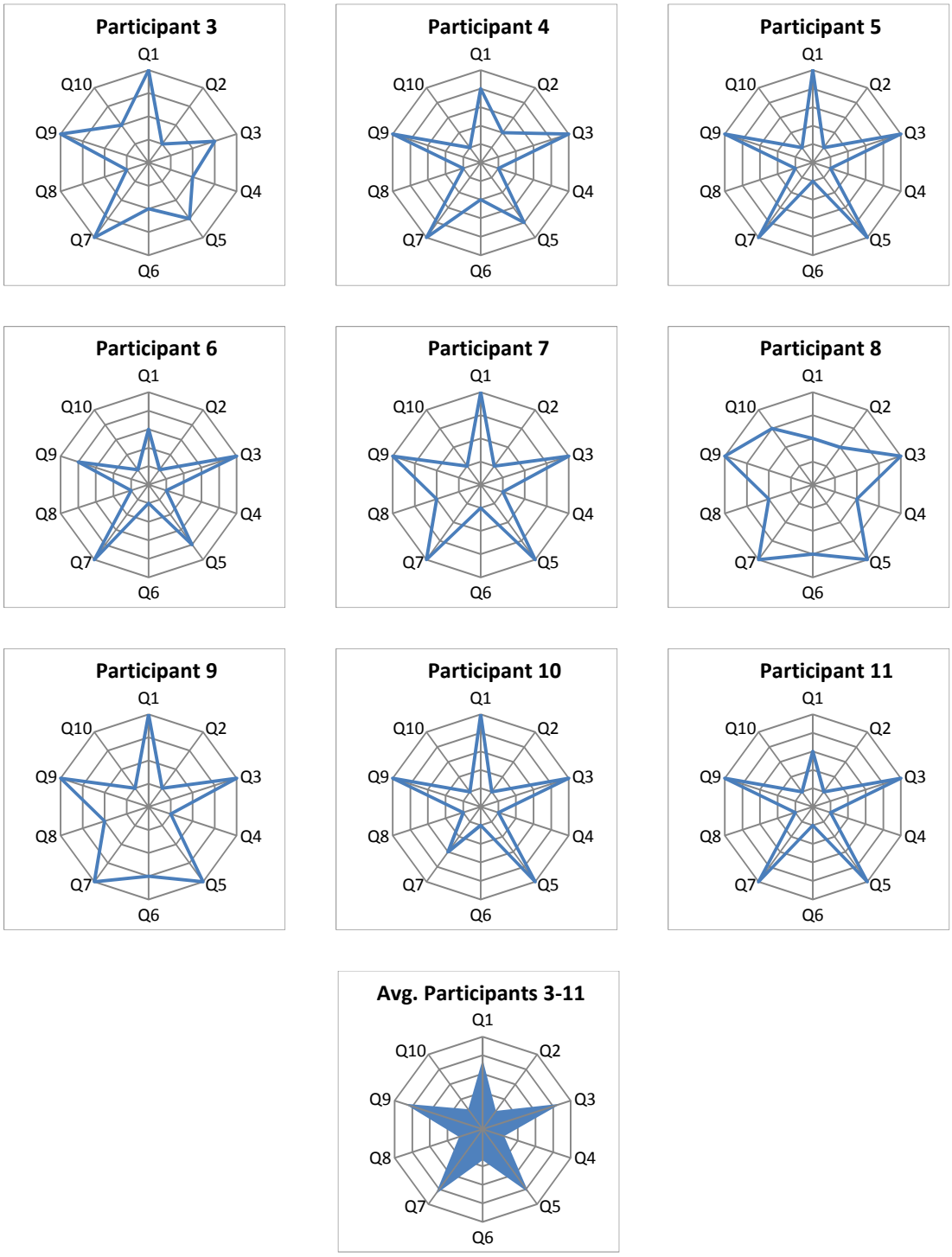

Figure 20 Individual SUS scores and SUS averages for the end-user-interface, main-study

5.5.2 Algorithm Performance

5.5.2.1 Rankings of the First Setup (No Sentiment Consideration)

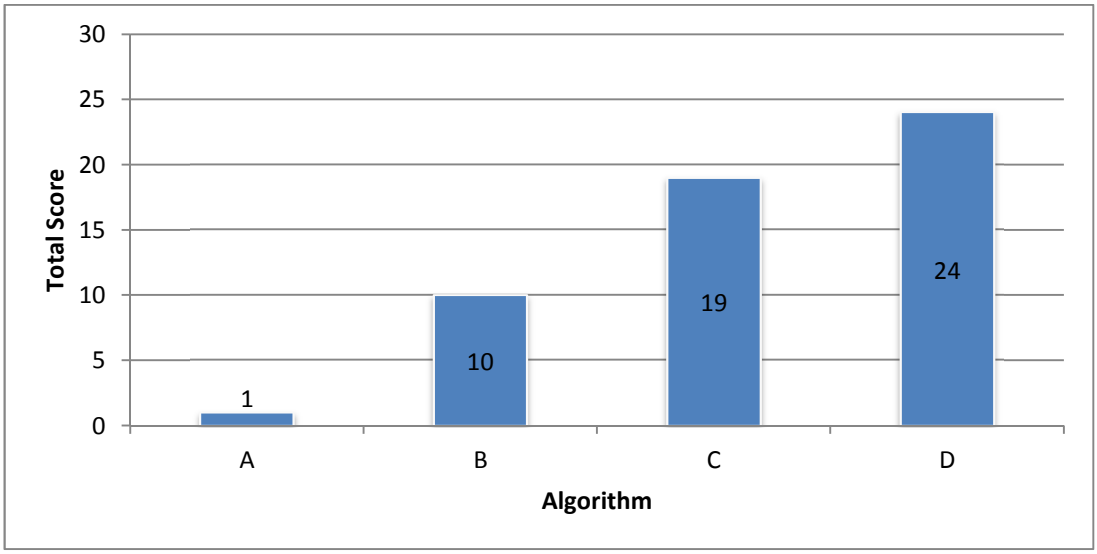

Figure 21 Total Score of Algorithms with no Sentiment Consideration (First Setup)

In the first setup, most users favoured column D overall, which is the combined algorithm (expert-enhanced and social-signal enhanced). The random expert-enhanced algorithm C was close with just 5 points less. Many users expressed that there was a clear separation between columns C and D and columns A and B.

5.5.2.2 Rankings of the Second Setup (Sentiment Positive or Negative)

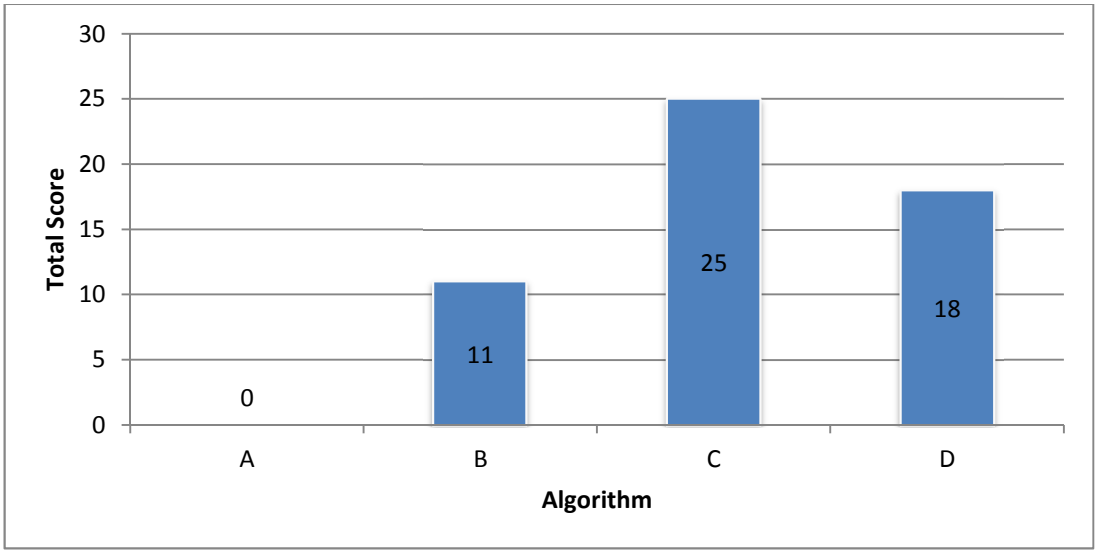

Figure 22 Total Score of Algorithms with Sentiment Positive or Negative (Second Setup)

In the second setup, which only displayed entries with positive or negative sentiment, the columns C and D still outperformed A and B, though column C was rated higher than D in this setup.

5.5.2.3 The Impact of Sentiment on the Perceived Relevance

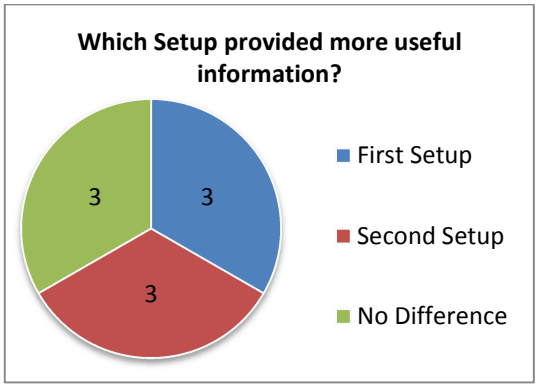

Figure 23 Perceived Quality on Sentiment Enabled/Disabled Setups

The participants did not perceive a difference when enabling the sentiment filtering for the content overall. Even though six people stated that they found setup one or two more useful, in conversations it became clear that this difference was very marginal and none of the participants favoured either setup over the other one strongly.

5.5.3 Overall Usefulness and Satisfaction with Selected Entries

5.5.3.1 Q16: The information from Twitter is useful and improves my knowledge about the product

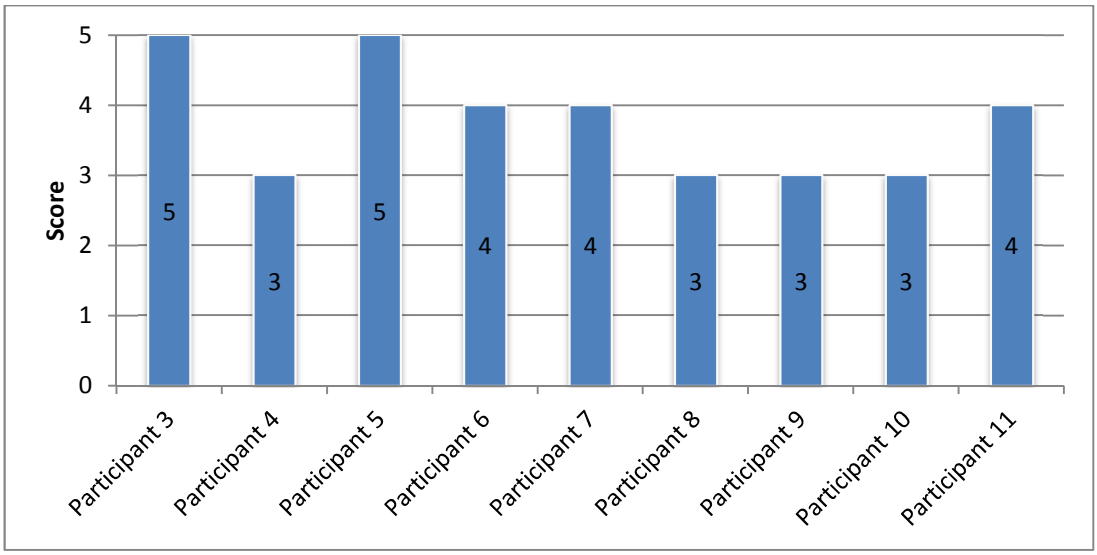

Figure 24 Q16, Usefulness of Provided Information Overall

With an average score of 3.77, most participants agreed that the information overall was useful and improved their knowledge about a product.

5.5.3.2 Q17: The displayed entries were relevant to the product

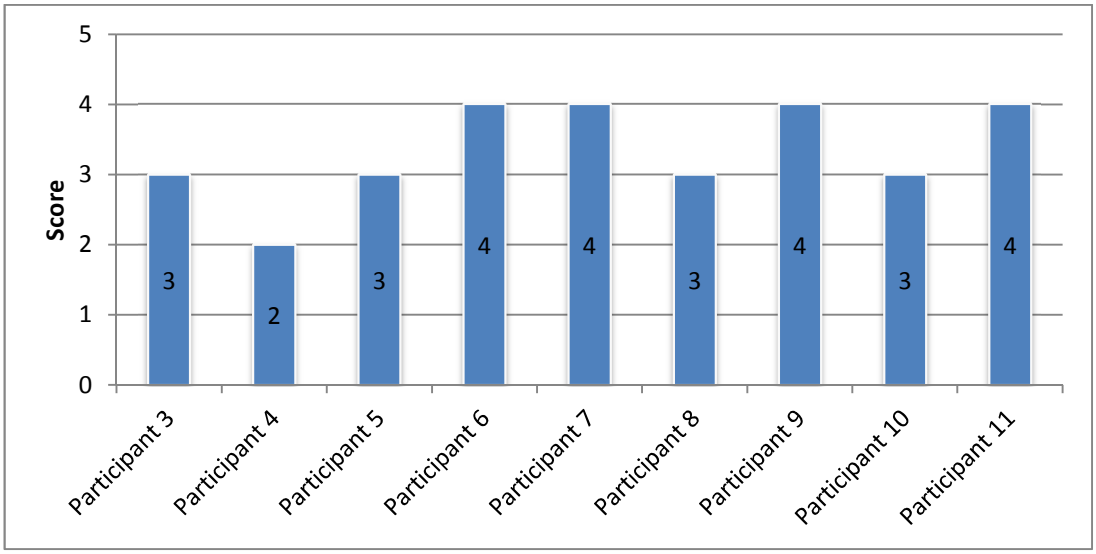

Figure 25 Relevance of Displayed Entries

Figure 25 shows the individual scores for question 17, with an overall average score of 3.33. The score was only average, mostly because it became clear that it is important to have a clear goal when using such a system, this will be discussed in more detail in the summary of this chapter.

5.5.3.3 Q18: I would like to have more control over what is being displayed

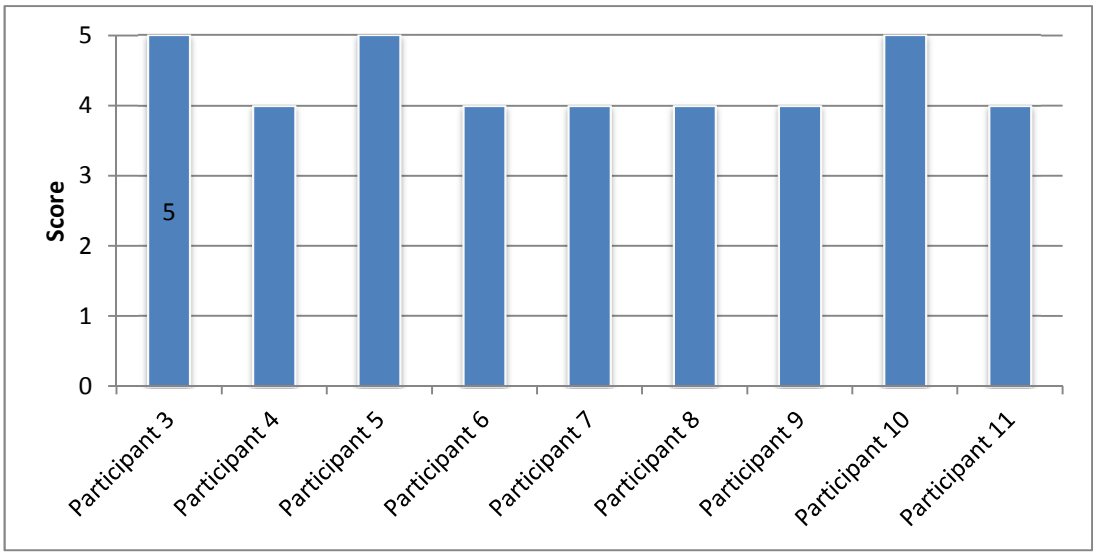

Figure 26 Desired Level of Control over the System

Question 18 resulted in the highest rate of agreement, with an average of 4.33: all participants would like to have more control over what is being displayed. The presented system just allowed the users to navigate through the product pages and to refresh a page. There were no options for filtering or slicing the displayed data. At this point of the experiment the user was not aware of the expert-view which was going to be tested next and would allow to have more control over the content.

5.5.3.4 Q19: I would use this system when considering the purchase of electronic devices

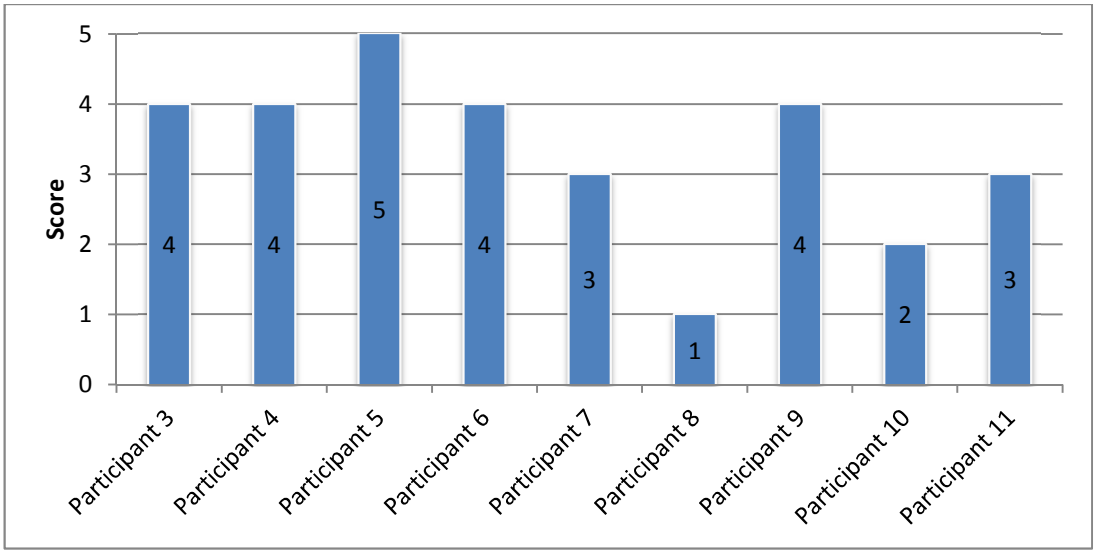

Figure 27 Anticipated Future use of the System

Most participants would like to use the Peregrine system to gain knowledge when considering the purchase of electronic devices. The participants with neutral or disagreeing ratings mostly stated that they generally do not prefer the format of Microblogging technologies and rather go to other sources and review sites.

5.5.4 Observations and Feedback from the Participants

Similar to the pilot-test, many participants tried to come up with a methodology to rate the columns with generated content. Most people took at least some notes to remember the preferences of each setup and to decide on an overall ranking.

Many participants stated that they would like to have more control on what is being displayed. This is a reasonable feedback, but due to the limitations of Twitter, receiving and storing entries in real-time rather than accessing a repository of past entries is much more feasible.

It was also interesting to observe that many participants tried to find patterns in the kind of content that is being displayed in each column. For instance, some users had the perception that one column is more about user experiences while another is more about news on the product. While this was not planned in the original algorithms, this is a natural result: The selected sources for the expert-enhanced algorithms C and D were mostly popular news sources, algorithms A and B on the other hand would naturally select more entries from casual users.

Two participants stated that they would like to see a classifier to improve the quality over time:

> "I would add a sort of Stackoverflow[50] styled interface to promote or downgrade the Tweets" – Participant 10

> "I would like to be able to rate the results in order to improve the systems usability with time" – Participant 9

This could be realised in many ways, e.g. with a simple Naïve Bayesian Classifier, and could indeed lead to major improvements in the quality of selected data.

[50] A popular website for software engineers to post questions and answers to technical problems, http://stackoverflow.com

5.6 Second Usability Test: Experts

5.6.1 SUS Scores

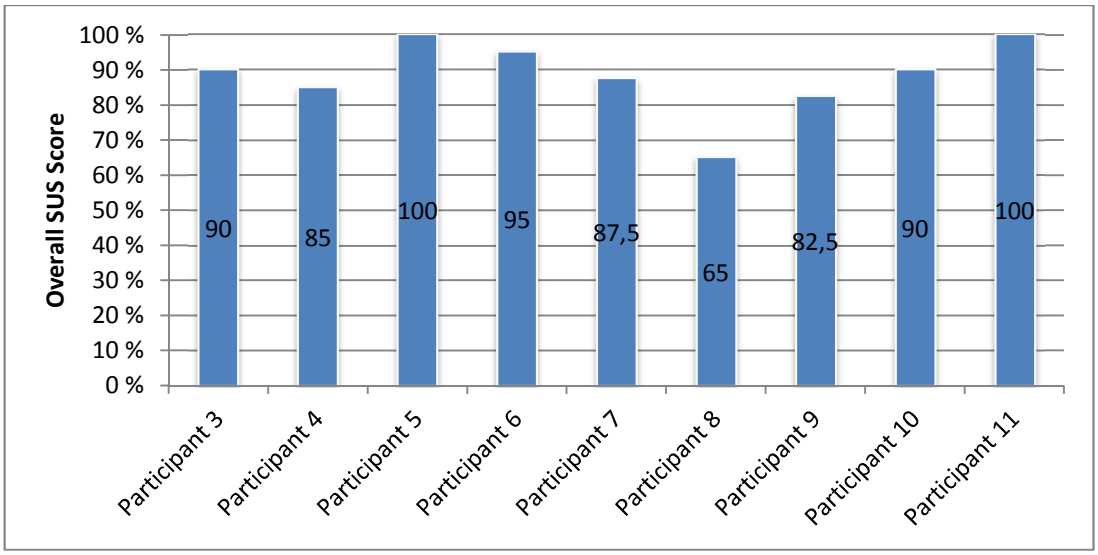

Figure 28 SUS-Averages for the expert- interface in the main-study

The SUS-Score average of the expert-interface to define and manipulate keywords and relevant authors of a domain was 88.33% and had a standard deviation of 10.07%. This high score was unexpected because many participants had a non-technical background and this task seemed rather technical. It turned out that the non-technical people did not have difficulties to understand what this user-interface is designed for and how to use it. Figure 29 shows the individual user scores and the overall average in form of a spider diagram.

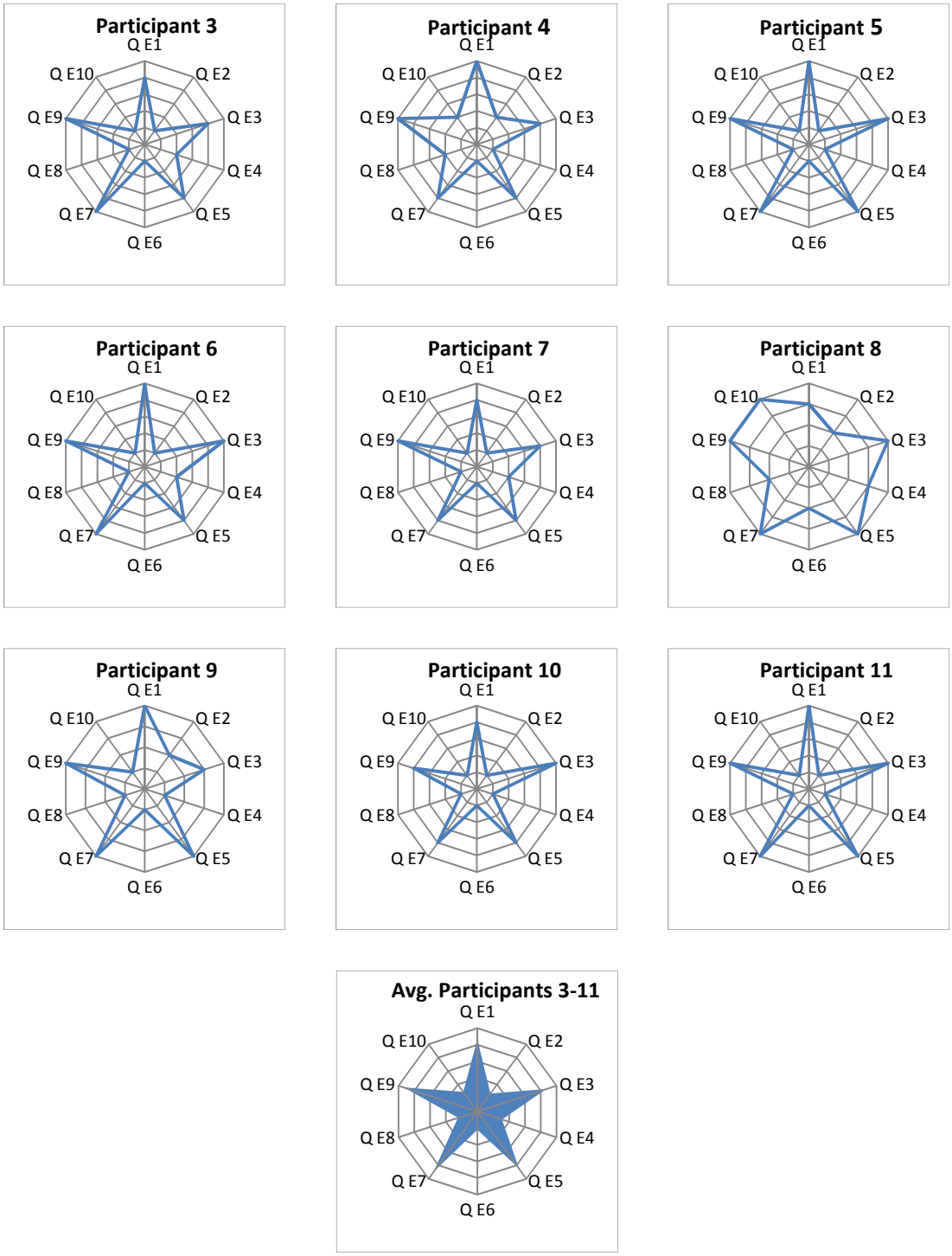

Figure 29 Individual SUS scores and SUS averages for the expert-interface, main-study

The averages show, similar to the end-user-interface evaluation, that most users agree on a high usability of this system.

5.6.2 Observations and Feedback from the Participants

It was interesting to observe that the purpose and the features of this system were accepted and understood quickly. Even though many participants did not know a single author on Twitter, the concept of defining interesting sources seemed to be no burden. The feature which automatically suggests authors from Twitter when a participant started typing in a box was recognised as particularly useful. The feedback from users was lesser than in the first system, though some participants stated that they would like to have a better introduction of what is being performed with this system, and more explanatory descriptions.

5.7 Summary of Evaluation

The goal for this research was to address the question if a system can be developed to utilise structured sources of information in order to better understand content derived from many users. Further, it was envisaged that by using expert-enhanced annotations and social signals to create tailored views on Social Media content, the perceived quality and the relevance of selected entries can be improved significantly. To address these problems, four algorithms to select entries from Social Networks were developed. Participants were asked to rank the selected content by these algorithms to reflect the level of relevance. Further, an expert-system to define semantics in order to collect Social Media content was developed. The participants were asked to annotate their interests of a domain in this system.

The results of this evaluation illustrate the feasibility of developing a system to support these goals. DBpedia was utilised to successfully describe a domain and to help with the process of selecting relevant information. The majority of the participants agreed that the Peregrine system provides information which increases their knowledge about the product (Q16). The high SUS scores for the end-user interface and the expert-interface proof that the participants felt comfortable using Peregrine. The participants found the entries relevant and useful and would like to use a similar system when considering the purchase of electronic devices. Most of the users would like to have more control over what content is being displayed.

5.7.1 Experts vs. End-users

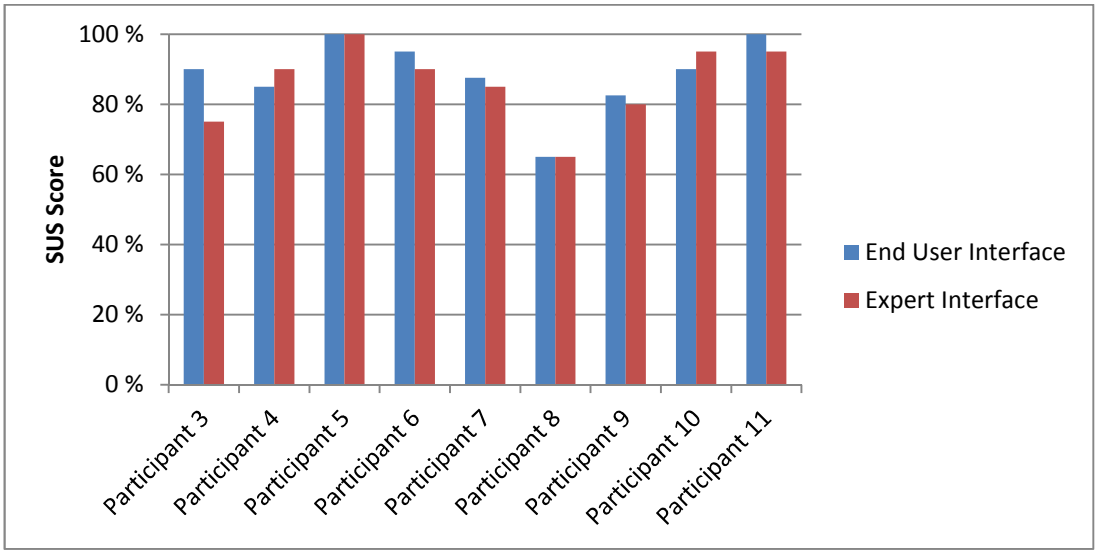

Figure 30 Comparison of the SUS Scores of Both Experiments

The expert-interface was highly accepted and the participants would be interested in seeing the results of their input to this interface. In the beginning of this work it was envisaged that there is a distinction between end-users and experts. The results of this experiment led to the conclusion that it is feasible for both casual users and advanced users to create their own filters and to build tailored views, even with little knowledge about the characteristics of the Social Network. Further, there is a high correlation between the ratings for the expert-interface and the end-user interface, illustrating the level of knowledge necessary to use both systems does not vary very much.

5.7.2 Algorithms and Social Signals

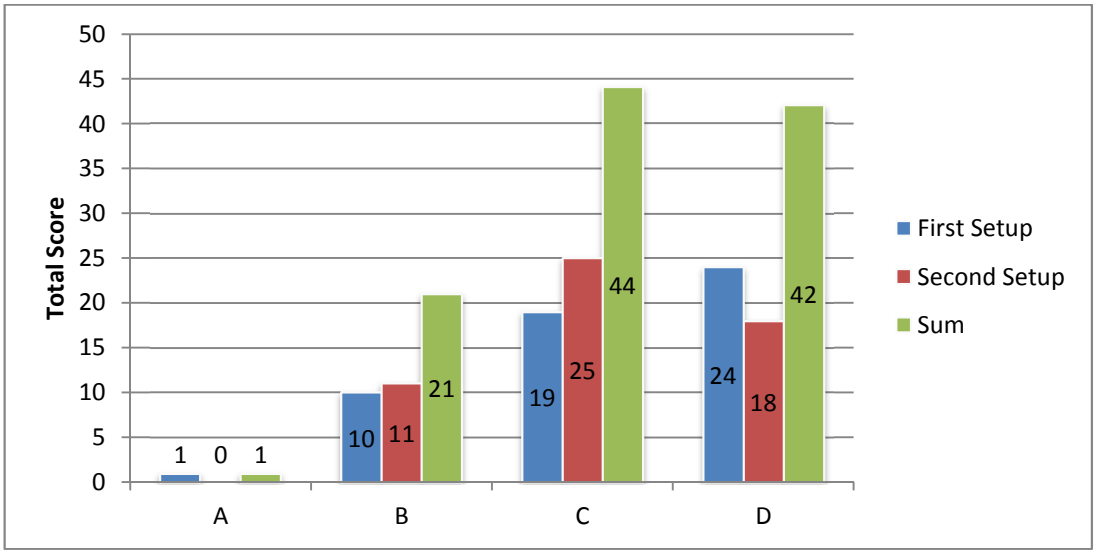

Figure 31 Algorithm performance of Different Setups

Evaluating the algorithms led to a clear result, favouring the generated content based on an expert-enhanced set of relevant authors. Columns C and D outperformed columns A and B by far: combined they gained 86 points while A and B only gained 22 points. While the columns C and D where ranked much higher in global, the participants ranked C only slightly higher than D.

This shows that the social signals improved the performance of algorithm B significantly in contrast to the random selection of algorithm A, but the impact of social signals in the expert-enhanced algorithms C and D was negligible. The quality of the entries was raised by the expert-enhanced selection of trusted authors to a degree that the social status did not have a high enough impact on the selection to cause a noticeable change for the participants. However, the impact on the perceived quality of algorithm B shows the highest increase in quality overall.

The second setup, which only used positive or negative entries, performed similar to the first setup. Column C was rated slightly higher than D in the second setup, while column D outperformed C in the first setup. However, when asking the participants if one of the setups delivered better results overall, the answers were distributed equally; hence no significant difference was observed overall. The consideration of the sentiment did not lead to a conclusive result.

5.7.3 Conclusion of Evaluation

An issue with the Peregrine system was caused by using the term “relevance” as key criteria for ranking the algorithms. For instance, many participants felt that it would make a difference if they are looking to purchase a device or if they already own a device. Relevance always depends on what a person is looking for. If the concept of this project would be expanded further to a system to select relevant content from any domain, this problem would become even more complex. It is non-trivial to design a system to select entries based on a certain goal, especially in a generic context. However, in Social Networks often there is no clear goal of what to achieve: the richness of these projects comes from the exploration of the provided information through following relationships and discovering new interests and pieces of content.

Only one social signal was selected (the number of followers) and improved the results compared to a random selection of data significantly. Many other signals have been proposed and were considered as attributes for the algorithms. Some have not been selected due to technical limitations (such as the number of Retweets), but overall a single strong signal was good enough to select very relevant content.

In contrast to other research, which often only focuses on statistics of Social Networks, this work allows to observe the impact of a social signal. Also, the correlation between social signals, expert-enhanced annotations of a domain and the impact of sentiment analysis has not been analysed in the same setup before. Unlike the work of P.A. Gloor et. Al. [8], this work only uses signals that exist in Social Networks, rather than utilising third-party technologies such as Google PageRank, which does not give detailed insight on how it is ranking pages.

6 Conclusion and Future Work

6.1 Achieving the Research Aims

The aim of this research was to address the problem of Social Network exploration which has become difficult in the past, mostly due to the vast amount of content produced, the nature of multithreaded communications and very little semantic knowledge about the content. This work addressed the question to what degree sentiment analysis, social signals, the utilisation of external semantics and expertise can improve the relevance of selected Social Media content.

Specifically, the following main research aims were identified in chapter 1:

- To develop a system to retrieve Social Media data, to cache it and to make it accessible in an efficient matter
- To investigate in tools and techniques to identify the subject of short pieces of content and to reduce noise, spam, duplicates and partial duplicates
- To use open knowledge initiatives, such as Linked Data, to help categorise unstructured content and to identify related keywords to a domain
- To identify the signals which have the biggest impact of the relevance of such entries
- To investigate in tools to determine the sentiment of the collected data and to measure their impact on the relevance
- To visualise identified pieces of information and to evaluate the quality of the presented data

A total of 2.8 million Tweets was downloaded and used as input for the developed Peregrine system. To maintain a high performance when traversing millions of records, database optimisations and a decrease of the size of required entries were performed. Further research in caching structures to efficiently access Social Media content based on personal preferences would be beneficial.

To identify the subject of short pieces of content, DBpedia was utilised to gather keywords related to the topics of interest. One key challenge of this implementation was the detection of partial duplicates which occurred very frequently. This was addressed by utilising the Levenshtein distance algorithm which successfully eliminated redundant

content. To reduce the level of noise and spam, techniques to compare the date of the publication and the date of the account-creation of the author were successfully implemented.

DBpedia was utilised to make suggestions of keywords that describe a knowledge domain. The information provided by DBpedia proved to be slightly incoherent at some points. For instance, when looking at different companies and how relations to their products are semantically described in DBpedia, very different structures and keywords were used. To expand this work to support other knowledge domains, further research in mapping to DBpedia and other sources of structured information will be required.

Multiple signals from within and outside of the Social Network have been considered to rank the selected content. Four different algorithms to reflect different signals and their correlation have been implemented. The best results were achieved by an expert-enhanced signal: a defined set of trusted authors. The perceived relevance was the highest when only displaying entries by authors of this created set. Further, the social signal "number of followers" in Twitter significantly improved the perceived relevance compared to a random selection of data.

To address the aim of utilising sentiment analysis to improve the results, the sentiment of Social Media was gathered by utilising the Alchemy API. The selected content was presented to the participants of the evaluation in two different setups:

1. Any qualified content was displayed, regardless of its sentiment
2. Only qualified content with a positive or negative sentiment was displayed

The underlying motivation was the hypothesis that setup 2, possibly displaying polarising content, might lead to better results. The participants were asked to rank the content selected by the algorithms and to give feedback if overall setup one or setup two presented more relevant content. The results of the sentiment-enhanced experiment were not conclusive: Three participants stated that setup 1 showed better content; three participants preferred setup 2; and three participants did not notice a difference between the setups.

Finally, to create an appealing visualisation, state of the art technologies were used and two user-interfaces were built. Firstly, the end-user interface which displayed Social Media

content according to the defined semantics and algorithms. Secondly, an expert interface was created which allows the annotation of a domain in an efficient matter. Ajax technologies combined with the dynamic presentation of suggestions from external sources (DBpedia for related keywords and Twitter when searching for authors relevant to a domain) resulted in very high usability scores in the evaluation with potential users.

6.2 Contribution

The results of the evaluation proved that the use of social signals significantly improv the perceived relevance of the entries in contrast with a random selection. The expert-enhanced annotations further improve the results in contrast to the random selection and of the social-signal based selection. Many possible social signals were evaluated, eventually only one social signal was used and had a strong impact on the quality of the selected content.

The usability of the Peregrine system to annotate a knowledge domain based on suggestions from DBpedia as well as the output of the selected Social Media content was ranked high with averages over 85%. This suggests that the separation between experts and end-users is not very distinct but rather a blurring line; a system may be developed which combines these two user-interfaces and allows a dynamic adaption and tailoring to personal preferences.

This work illustrated the scope of this problem and suggests that selecting relevant content from Social Networks is a challenging problem. However, the results of the evaluation show that this work has significantly improved the exploration of Social Media content and allows an effective traversing of information derived from many users. Finally, there is a strong potential for further research and experiments in the area of expert mediated ranking of Social Media content.

6.3 Future Work

This work raises a number of new challenges and creates space for further research in the following areas:

1. *Integration of the user's social graph:* When designing a system to create tailored views for casual end-users, a lot of preferences, social relationships and topics he/she is interested in exist already in Social Networks. For instance, the Facebook API allows

accessing the personal information about a user, which could be utilised to make suggestions of keywords and content, in addition to the suggestions from DBpedia.

2. *Expansion of the scope to enhance the exploration of multiple Social Networks simultaneously:* Currently, this work only uses data provided by Twitter, mostly due to its nature of public communication and data availability. Other Social Networks exist and often capture valuable information and expert-knowledge. Future work might support the exploration of multiple sources for Social Media at the same time.
3. *Utilisation of more signals:* In this work, only very few signals have been used to categorise and rank content. While this approach proved to be very effective and the quality of the selected data was rated much higher than a random selection, research in further signals may improve the quality of the selected content.
4. *Real-time updates:* The implementation presented in this work gathered Social Media content and built a data-corpus which was used to select relevant information. Further work might take into account that data is being published rapidly and relevant content could be added to the data-corpus and the user-interface in real-time.
5. *Advanced user-interfaces:* The results of the user-evaluation led to the conclusion that there is no clear distinction between experts and end-users. Most participants felt comfortable using both systems and many stated that they would prefer to have more control over what content should be displayed. Therefore it would be beneficial to merge the annotation of a knowledge domain with the output of the selected Social Media content.

Appendix A – Expert-enhanced Content

id	name
1	IPod
2	IPhone
3	IPad

Table 8 Selected Products for the Twitter Crawler

keyword	product_id
16gb	2
16gb	3
32gb	2
32gb	3
8gb	2
8gb	3
appstore	2
Appstore	3
Audio Player	1
Digital audio player	2
FaceTime	2
FaceTime	3
HD Video	2
HD Video	3
headphones	1
icloud	2
Icloud	3
Industrial Design	1
iOS	2
iOS	3
Ipad	2
Iphone	3
ITunes	1
Itunes	2
ITunes	3
media player	1
Microphone	1
Microphone	2
Microphone	3
Mobile phone	2
mp3	1
mp3	2
mp3	3
Multi-touch	1

Multi-touch	2
Multi-touch	3
Multitouch	3
Music	1
Music Player	1
Nano	1
personal digital assistant	2
portable	1
Portable	3
Retina	2
running	1
Shuffle	1
Smart Cover	3
Smartphone	2
Tablet	3
Touch	1
Touchscreen	1
Touchscreen	2
Touchscreen	3
Wi-Fi	3
Wi-Fi device	2

Table 9 Selected Keywords for the Twitter Crawler

screen_name
9to5mac
appleinsider
arstechnica
CNET
computerworld
elreg
ENGADGET
everythingicafe
GIGAOM
GIZMODO
guardiantech
iPhoneAlley
jeffjarvis
Lifehacker
MacRumors
MacRumors
mashable
nytimesbits
nytimestech
oreillymedia
pcmag

pcworld
siliconrepublic
Slashdot
TechCrunch
Techmeme
techradar
TelegraphTech
TiPb
TNWapple
tnwgadgets
TNWmobile
tomshardware
TUAW
ubergizmo
wired
ZDNET
zipadblog
zipadblog

Table 10 Selected Authors for the Twitter Crawler

Appendix B – Collected Data from Questionnaires

In this appendix, the results of the evaluation are presented in detail. In all tables the columns represent the 11 participants. Chapter 5.3 presented details on the questionnaires and the metrics.

	1	2	3	4	5	6	7	8	9	10	11
Q1	3	3	4	4	5	3	4	2	4	5	3
Q2	1	1	1	2	1	1	1	2	1	1	1
Q3	5	5	3	5	5	5	4	4	4	5	5
Q4	1	1	2	1	1	1	1	2	1	1	1
Q5	4	5	3	4	5	4	4	4	4	5	5
Q6	1	1	2	2	1	1	1	3	3	1	1
Q7	5	5	4	5	5	5	4	4	4	3	5
Q8	1	1	1	1	1	1	2	2	2	1	1
Q9	5	5	4	5	5	4	4	4	4	5	5
Q10	3	1	2	1	1	1	1	3	1	1	1

Table 11 SUS Scores for the end-user interface

	1	2	3	4	5	6	7	8	9	10	11
Q1	3	4	4	5	5	5	4	3	4	4	5
Q2	1	1	1	2	1	1	1	2	2	1	1
Q3	5	5	4	4	5	5	4	4	3	5	5
Q4	4	1	2	1	1	2	2	3	1	1	1
Q5	4	5	4	4	5	4	4	4	4	4	5
Q6	1	1	1	1	1	1	1	2	1	1	1
Q7	5	5	5	4	5	5	4	4	4	4	5
Q8	1	1	1	2	1	1	1	2	1	1	1
Q9	2	5	5	5	5	5	5	4	4	4	5
Q10	4	1	1	2	1	1	1	4	1	1	1

Table 12 SUS Scores for the expert interface

	1	2	3	4	5	6	7	8	9	10	11
Q11	N	N	IU	IU	IU	R	IU	N	N	N	N
Q12	B.Sc.	LLM	B.Sc.	B.Sc.	B.A.	B.Sc.	B.Sc.	B.Sc.	B.Eng.	B.Sc.	M.A.
Q13	Female	Female	Male	Male	Male	Male	Male	Female	Male	Male	Male

Table 13 Questionnaire answers Q11-Q13

(IU = Infrequent User, N = Never used it, R = Regular user)

	1	2	3	4	5	6	7	8	9	10	11
Q14a	D, C, B, A	C, D, B, A	D,C,B,A	D,C,B,A	C,B,D,A	D,C,B,A	D,C,B,A	D,C,B,A	D,B,C,A	D,C,A,B	C,D, B,A
Q14b	D, C, A, B	D, C, B, A	D,C,B,A	C,D,B,A	C,D,B,A	C,D,B,A	D,C,B,A	C,B,D,A	C,B,D,A	C,D,B,A	C,D,B,A
Q15	No difference	Setup 2	Setup 2	Setup 2	Setup1	No difference	Setup 1	No Difference	Setup 1	No Difference	Setup 2

Table 14 Questionnaire answers Q14 and Q15

(a = no sentiment consideration, b = only positive or negative entries)

	1	2	3	4	5	6	7	8	9	10	11
Q16	4	2	5	3	5	4	4	3	3	3	4
Q17	4	4	3	2	3	4	4	3	4	3	4
Q18	5	3	5	4	5	4	4	4	4	5	4
Q19	5	3	4	4	5	4	3	1	4	2	3

Table 15 Questionnaire answers Q16-Q19

References

[1] J. Breslin and S. Decker, "The Future of Social Networks on the Internet," *Internet Computing, IEEE*, pp. 86 - 90, 11-12 2007.

[2] Rongjing Xiang, Jennifer Neville, and Monica Rogati, "Modeling relationship strength in online social networks," in *Proceedings of the 19th international conference on World wide web*, 2010.

[3] P. Mika , "Flink: Semantic Web technology for the extraction and analysis of social networks," in *Selcted Papers from the International Semantic Web Conference*, 2004.

[4] A. Sheth, "Citizen Sensing, Social Signals, and Enriching Human Experience," *Internet Computing, IEEE* , vol. 13, no. 4, pp. 87-92, 07-08 2009.

[5] Meenakshi Nagarajan et al., "Spatio-Temporal-Thematic Analysis of Citizen Sensor Data: Challenges and Experiences," in *Lecture Notes in Computer Science*.: Springer-Verlag Berlin, 2009, pp. 539-553.

[6] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a Social Network or a News Media?," in *19th International World Wide Web Conference*, 2010.

[7] Meeyoung Cha and Hamed Haddadi and Fabricio Benevenuto and Krishna P. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy," in *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media (ICWSM)*, Washington DC, USA, 2010.

[8] P.A Gloor, J. Krauss, S. Nann, K. Fischbach, and D. Schoder, "Web Science 2.0: Identifying Trends through Semantic Social Network Analysis," in *International Conference on Computational Science and Engineering*, 2009.

[9] Peter A. Gloor and Zhao Yan, "TeCFlow – A Temporal Communication Flow Visualizerfor Social Network Analysis," *ACM CSCW Workshop*, 2004.

[10] S. Siersdorfer, J. San Pedro, S. Chelaru, and W. Nejdl, "How useful are your comments?

- Analyzing and Predicting YouTube Comments and Comment Ratings," in *9th International World Wide Web Conference (WWW2010)*, 2010.

[11] A. Passant, P. Kapanipathi, and P. N. Mendes, "Twarql: tapping into the wisdom of the crowd," in *Proceedings of the 6th International Conference on Semantic Systems*.

[12] P.N Mendes, A. Passant, P. Kapanipathi, and A.P. Sheth, "Linked Open Social Signals," in *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, Toronto, ON, 2010, pp. 224 - 231.

[13] S. Auer et al., "DBpedia: A Nucleus for a Web of Open Data," in *6th International and 2nd Asian Semantic Web Conference*, 2007.

[14] V. Wade, C. Bruen, M. Gargan O. Conlan, "Multi-Model, Metadata Driven Approach to Adaptive Hypermedia Services for Personalized eLearning," in *Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, Malaga,Spain, 2002.

[15] A. Kobsa and W. Nejdl P. Brusilovsky, *LNCS: The Adaptive Web, Methods and Strategies of Web Personalization*.: Springer, 2007.

[16] P.Brusilovsky, "Knowledgetree: A distributed architecture for adaptive e-learning," in *Proceedings of the 13th international World Wide Web conference*, 2004.

[17] W. Hall, D. Roure M. Sah, "Dynamic linking and personalization on web," *SAC*, pp. 1404-1410, 2010.

[18] K. Koidl, O. Conlan, and V. Wade, "Non-Invasive Adaptation Service for Web-based Content Management Systems," , Torino, Italy, 2009.

[19] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing* , 2002.

[20] T. Berners-Lee, "Linked Data," *International Journal on Semantic Web and Information Systems. W3C*, 2006.

[21] John M. Trenkle William B. Cavnar, "N-Gram-Based Text Categorization," in *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, 1994.

[22] Thomas Tullis and William Albert, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*.: Morgan Kaufmann, 2008.

[23] J. Brooke, "SUS: a 'quick and dirty' usability scale," in *Usability Evaluation In Industry*., 1996, pp. 189-194.

[24] T.S., & Stetson, J.N. Tullis, "A Comparison of Questionnaires for Assessing Website Usability," in *Proceedings of UPA*, 2004.

[25] O. Conlan C. Hampson, "Leveraging Domain Expertise to Support Complex,Personalized and Semantically Meaningful Queries Across Separate Data Sources," *The IEEE Fourth International Conference on Semantic Computing*, 2010.

[26] A. Angehrn, and A. Maedche L. Razmerita, "Ontology-based user modeling for knowledge management systems," *User Modeling The 2003, ser. Lecture Notes in Computer Science*, 2003.

[27] D. Dagger, V. Wade, and O. Conlan, "Towards a standards-basedapproach to e-learning personalization using reusable learningobjects," in *Proc. of World Conference on E-Learning*, 2002, pp. 210-217.

[28] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: real-time event detection by social sensors," in *Proceedings of the 19th international conference on World wide web*, 2010.

[29] V. Geroimenko, *Visualizing the Semantic Web*, 2nd ed.: Springer, 2004.

[30] A. Passant, T. Hastrup, U. Bojars, and J. Breslin, "Microblogging: A Semantic Web and Distributed Approach," *4th Workshop on Scripting for the Semantic Web*, 2008.

[31] S. Kim and E. Hovy, "Determining the sentiment of opinions," in *Proceedings of the 20th international conference on Computational Linguistics*, 2004.

[32] J. Gaugaz, J. Zakrzewski, G. Demartini, and W. Nejdl, "How to Trace and Revise Identities," in *The Semantic Web: Research and Applications, LNCS*.: Springer, 2009.