Computational Linguistics in the Netherlands Journal 3 (2013) 121-134 Submitted 06/2013; Published 12/2013

Dealing with big data: The case of Twitter

Erik Tjong Kim Sang* ERIKT@XS4ALL.NL
Antal van den Bosch** A.VANDENBOSCH@LET.RU.NL

*Netherlands eScience Center, Amsterdam, The Netherlands

**Radboud University Nijmegen, Nijmegen, The Netherlands

Abstract

As data sets keep growing, computational linguists are experiencing more big data problems: challenging demands on storage and processing caused by very large data sets. An example of this is dealing with social media data: including metadata, the messages of the social media site Twitter in 2012 comprise more than 250 terabytes of structured text. Handling data volumes like this requires parallel computing architectures with appropriate software tools.

In this paper we present our experiences in working with such a big data set, a collection of two billion Dutch tweets. We show how we collected and stored the data. Next we deal with searching in the data using the Hadoop framework and visualizing search results. In order to determine the usefulness of this tweet analysis resource, we have performed three case studies based on the data: relating word frequency to real-life events, finding words related to a topic, and gathering information about conversations. The three case studies are presented in this paper.

Access to this current and expanding tweet data set is offered via the website `twiqs.nl`.

1. Introduction

There is a growing interest from many different areas of research in using large text corpora for research. A number of projects try to meet these needs at a large scale. For instance, the Google Books project has digitized tens of millions of books and allows them to be searched on `books.google.com`, and Mark Davies offers data based on 1.9 billion words on `corpus.byu.edu`. There are also initiatives for Dutch such as for example the Nederlab project which was started in 2013 (`nederlab.nl`).

One of the text data sources which is gaining popularity for researchers as well as journalists, government officials, and policy makers is Twitter,[1] a social network on which people communicate by posting short text messages. An estimated one million of Dutch speaking people use the social network, resulting in millions of Dutch messages every day. Most of these current messages are publicly available and together constitute an interesting data resource for studying a wide variety of topics.

While Twitter allows searching through recent messages, the search results generate questions which are difficult to answer. For example, if we find forty messages about a certain topic on a given day, is this an unexpectedly large number? What was the total number of messages on that day? How does the coverage of the topic compare to yesterday, last week, last month or even last year? Is there interesting metadata associated with the search results?

These additional questions could be answered if we had Twitter messages available as a corpus. Attempts have been made to collect and distribute tweets for research purposes, such as Petrović et al. (2010) but these have been prohibited by Twitter. In its document *Developer Rules of the Road*, the company states that developers *will not attempt (...) to: (...) redistribute (...) access to (...) Twitter Content to any third party (...) you may only return IDs (including tweet IDs and*

1. `twitter.com`

user IDs) (Twitter Inc. 2013). In practice this restricts Twitter research to tweets one has collected oneself.

This restriction makes it difficult for people lacking strong computer skills or support to analyze large volumes of tweets. Our current work is aimed at making it easier for such users to get access to information about tweets. We aim at making available information about tweets written in Dutch to researchers without violating the Twitter developer rules. Our work deals with four questions. The first is what useful tweet information can be made available without making the tweets themselves available. The second question is how we can deal with the Twitter data volumes, which are large even if we restrict ourselves to Dutch tweets. The third question is how the information should be presented to the user. And the fourth question deals with the user feedback: is the restricted information we offer valuable for users?

This paper proceeds with a section on related work. Next we will describe our approach for selecting and collecting tweets, as well as searching, processing and visualizing information about tweets. After this we present three examples of how the resulting service can be employed by researchers. In the final section we make some concluding remarks.

2. Related work

In 2010, Petrović et al. (2010) presented the Edinburgh Twitter Corpus, a collection of 97 million tweets for research purposes. However, within two months after the publication they were asked by Twitter Inc. to stop distributing the corpus.

There have been numerous papers based on tweets before and after 2010, all using tweets collected by the authors, for example Tumasjan et al. (2010) and Liu et al. (2011). Tjong Kim Sang (2011) presented instructions for linguists about how to collect tweets as well as examples of how tweet information can be visualized.

At present three companies, certified data resellers of Twitter, offer access to all tweets: DataSift, Gnip and Topsy. Users can search in tweets and metadata, and receive search results in different formats. Unlike searching on `twitter.com/search`, the full access search offered by these companies is not free.

3. Data, problems and solutions

In this section we describe our methods for collecting tweets, identifying the language they are written in, searching collections of tweets, performing linguistic analysis on tweets, and visualizing tweet information.

3.1 Data collection

We collected billions of Twitter messages (tweets) with the filter part of Twitter's streaming API (`dev.twitter.com/docs/api/1.1/post/statuses/filter`). This software allows a continuous search of new tweets based on the keywords present in the messages or based on the names of the users that sent the messages. We are only interested in messages written in the Dutch language. We use two strategies for collecting such messages.

First we search for a selection of 229 Dutch words and hashtags (see Figure 1)[2]. Most of the words originate from a list of frequent Dutch words. Care has been taken to select words which are not frequently used in other languages. Still, non-Dutch messages regularly slip through this filter. The word list can easily be expanded. However, the current word list already selects more than the maximum of messages we may retrieve from the website (50 tweets per second). Adding extra words to the list would not increase the number of messages collected by this filter.

2. We also search for tweets containing Dutch dialect words but this feed produces only a few tweets per hour.

aan achter alleen allemaal alles als altijd alweer andere anders beetje beneden bent beter bij bijna binnen blij buiten #bzv daar daarna dacht dag dagen denk deze dingen dit doen doet dood #durftevragen dus #dutchteenagers #dwdd echt een eens eerst eerste egt eigenlijk eindelijk enzo erg eten gaan gaat gedaan geen #geenzin gehad gelijk gelukkig gemaakt geweest gewoon gezellig gezien ging goed #gtst gwn haar haat halen heb hebben hebt heeft heel hele helemaal hem het hier hij hoop hoor hou huis iedereen iemand iets ik infokunde informatiekunde jaa jaar jij jou jullie kamer kapot keer kijk kijken klaar komen komt krijg kunnen kut laat laatste laten lekker #lekker leren leuk leuke leven lief lieve maak maakt maandag maar maken meer mensen mij mijn misschien moeder moest moet moeten mooi mooie morgen naar niemand niet nieuwe #nieuws niks nodig nog nooit nou ofzo omdat onder ons onze ook op paar #penw #pownews praten #rtl7 rug schatje slaap #slajezelf slapen #slapen snel soms staan staat stad steeds straks tegen terug thuis #tienerfeiten #tienerthings tijd toch toen uit uur vakantie vanavond vandaag veel verder vind voel #voetbalfans volgende volgens voor vrij vroeg waar waarom wachten wakker wanneer weer weet weg wel wereld werk werken weten #widm wie #wiedoethet wij wil willen wilt worden wordt zal zaterdag zeg zeggen zegt zei zeker zelf zie zien zijn zin zit zitten zo zonder zou

Figure 1: List of the 229 keywords and hashtags used by the keyword filter for collecting Dutch tweets.

Our second strategy for collecting tweets is to gather all messages from a ranked list of 5,000 users who post messages in Dutch most frequently. This ranked list is updated each month based on all the messages we have collected in the previous month. This data stream does not reach the maximum of messages that can be retrieved. However, 5,000 is the maximum of users that may be tracked this way, which is substantially less than the estimated number of Dutch users on Twitter (about one million).

3.2 Language identification

The two tweet streams primarily contain Dutch tweets. However, we still found many foreign language tweets among them. In order to decrease the size of the latter group, we applied a language checker to the data. We chose libTextCat for this purpose, a language checker developed by Frank Scheelen from WiseGuys based on earlier work by Gertjan van Noord (`software.wise-guys.nl/libtextcat`). As extra support for the language checker, 3,004 tweets of which the checker was uncertain about the language, were manually inspected, annotated and used as extra training material for the checker. In this way we added some of the specific vocabulary of tweets to the identification process.

In order to evaluate the language identification process, we manually annotated a random sample of 1,000 tweets with language information and compared the language labels of the identification software with those of the human annotator. For comparison, we also evaluated a baseline system which assigns to a tweet the same language as the user has specified for his Twitter interface. The interface language information is available in the metadata of each tweet.

The two systems reached similar performance levels, shown in Table 1. We also tested two combinations of the systems. Selecting only tweets that were labeled as Dutch by both systems (AND) did not perform well, but selecting tweets that were identified by either of the systems (OR) performed very well. We adopted the second combination method for automatically identifying the language of the tweets in our collection. In the remainder of this work we only use the tweets identified as Dutch with the combined identification method.

	Found (in 1000)	Precision	Recall	$F_{\beta=1}$
Language identification	500 tweets	99.4%	68.3%	81.0
Interface language	542 tweets	97.6%	72.7%	83.3
Language identification AND interface language	362 tweets	100.0%	49.7%	66.4
Language identifcation OR interface language	680 tweets	97.6%	91.2%	**94.3**
Manual selection	727 tweets			

Table 1: Estimation of the performance of language identification based on a random sample of 1,000 of the 3.4 million tweets collected on Wednesday 22 May 2013. The language identification software finds almost the same number of Dutch tweets as using the Twitter language interface settings for selecting tweets (column *Found*). Selecting all tweets that were identified as Dutch by any of the two systems (OR) performs best when compared with the selection from a human annotator (columns *Precision*, *Recall* and $F_{\beta=1}$).

3.3 Searching billions of tweets

We want to search in our tweet collection and retrieve relevant tweets quickly. This task presents three challenges:

1. There is a lot of data: billions of tweets;
2. New tweets are constantly being added to the collection;
3. We want to be able to search in both tweet text and metadata.

Challenge 1 can be met by standard information retrieval systems. However, we do not know how these systems would cope with challenge 2 and challenge 3. We expected that the two remaining challenges would be hard for information retrieval systems and therefore we examined an alternative solution.

We chose the parallel Apache Hadoop environment[3] for searching in our tweet collection. By searching in different parts of the data in parallel, we could process large volumes of data in a reasonable time (challenge 1). Adding new tweets to the collection requires adding a new data file but no global index needs to be updated (challenge 2). Finally, the general architecture of Hadoop allowed for ad hoc search software being used, which enabled simultaneous search in text and metadata (challenge 3).

We were fortunate to be able to use the prototype Hadoop architecture of SurfSara, which is available to members of academic institutions in The Netherlands. The tweets were stored on the system in JSON format, the format distributed by Twitter. One extra key-value pair was added to each tweet, indicating the language of the tweet text according to our language guesser. Tweets were stored in compressed files (gzip) each containing one hour of data.

We implemented search software in the programming language Java. The software searches through one file (one hour of tweets) and returns tweets of which the text contains the required keywords or of which the metadata satisfies the provided metadata requirements (see Table 2). The Hadoop environment takes care of the parallelization if the user requires searching through more than one hour of data. Searching is not fast: searching for a word in a day of tweets requires about four minutes to complete. On the other hand, due to the parallelization, searching in larger time frames takes less than a factor of four minutes per se: about six minutes for a month of data and about half an hour for a year of data. We use a cache for retrieving searches which have been made earlier. The communication between the user and the system is performed by a separate program

3. `http://hadoop.apache.org/`

keyword	result
word	tweets containing "word"
$word_1$ $word_2$	tweets containing "$word_1$ $word_2$"
$word_1$, $word_2$	tweets containing "$word_1$" OR "$word_2$"
@user	tweets from user "user" and replies to these tweets
twinl-geo	tweets containing location coordinates
twinl-followers-min-number	tweets from users with at least "number" followers
twinl-followers-max-number	tweets from users with at most "number" followers
twinl-retweet	all retweets
twinl-gender-m	tweets from male users
twinl-gender-f	tweets from female users
twinl-age-18	tweets from users younger than 18 years
twinl-age-21	tweets from users between 18 and 25
twinl-age-26	tweets from users from 26 and older

Table 2: Available search instructions for tweet text and tweet metadata. While processing queries, the metadata features age and gender are guessed by software that uses a small set of hand-written rules.

which uses a web form and web visualizations, enabling anyone from anywhere in the world to access the system.

3.4 Linguistic processing of tweets

We expect that our Twitter search facility will be used by linguists (computational linguists, sociolinguists, theoretical linguists, etc.) some of whom may want to use word class information in search, such as in the query: *give me all tweets in which the word* table *is used as a verb*. In order to make such queries possible, we would like to add linguistic information to tweets. There are already some software packages available for automatically adding word class labels to Dutch text. However, these packages cannot easily be installed on the Hadoop system. Hadoop has its own file system structure and accessing software is different from the Linux systems that most of the language processing software is developed for.

Hadoop offers the possibility to run Java software. Unfortunately, the language processing software that we would like to work with, is not written in Java. As a test case, we developed our own Java version of a basic Dutch tokenizer, a program which separates words from punctuation characters. This tokenizer inserts white space before and after every character in the tweet that is neither a letter nor a digit. Because it was written in Java, it could be used as a part of the search software, processing search results before sending them to other parts of the system.

Our available project time did not allow for an implementation of the other required language tools in Java. An alternative approach would be to perform the language processing offline on Linux systems and store the results with the tweets. This is the approach we have taken for the identification of the language in which a tweet is written (see Section 3.2). However, storing the information with the tweets would increase the storage requirements. We are also worried about the additional processing requirements if these cannot be divided over multiple machines such as in the Hadoop framework, especially when processing the tweets we already have collected. Because of these two problems we have refrained from using the alternative of offline linguistic processing.

This means that we currently do not have a good solution for linguistic processing of our data collection. Searches that require linguistic information are currently not supported.

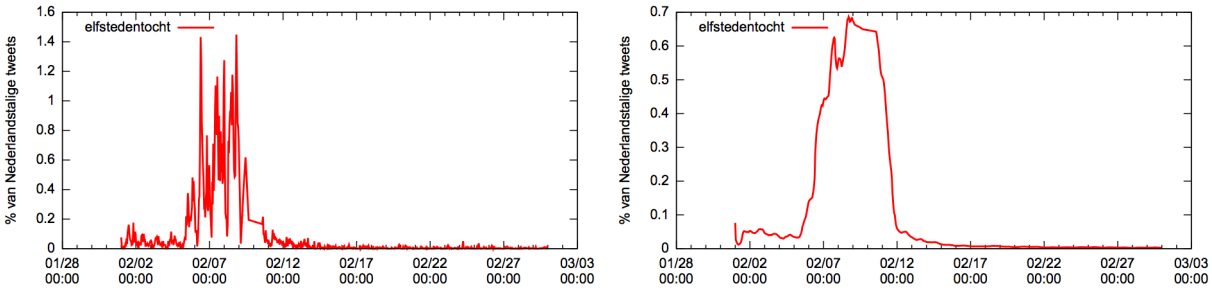

Figure 2: Two variants of the graph with the relative frequency of tweets containing the word *Elfstedentocht* (a Dutch speed skating race) in February 2013. The left graph has a smooth factor of 10 with graph points based on data from 10 minutes. The smoother one on the right has a smooth factor of 2000.

3.5 Visualization of search results

We are not allowed to redistribute the tweets (Twitter Inc. 2013) so we can only present summaries as search results We have chosen to present the following types of visualizations of the results:

1. A graph representing the relative frequency of the tweet results;
2. A map showing from which regions relevant tweets were sent;
3. A word cloud indicating words frequently used in the search results;
4. User information with gender and age distributions of the senders;
5. Tweet id information with the opportunity to inspect tweets at `twitter.com`

The graphs show relative frequencies rather than tweet counts because the number of tweets changes over the course of day. In the busiest hour (22:00-22:59) more than ten times as many tweets might be posted than in the most quiet hour (05:00-05:59). This daily rhythmic effect would be visible in any graph that displays tweet counts.

In order to be able to draw a graph, one has to choose the size of the time frame that the graph points are based on. In our graphs, the smallest time frame is one minute. The user may select any larger amount as time frame used for the data corresponding with one graph point. The default value is equal to the length of the search time frame in minutes divided by 144, with a minimum value of five. This corresponds to 10 minutes for a search time frame of 24 hours.

Figure 2 shows two versions of the same graph drawn based on different time frames. The left one uses ten minutes of data to determine the position of one graph point (smoothing factor 10). The right one uses data of 2000 minutes for the point data (smoothing factor 2000). Graphs based on a larger smoothing factor are generally smoother than the ones based on smaller smoothing factors. However, in graphs with a large smoothing factor small scale effects can disappear. What the best smoothing factor is for a graph depends on the user's interest.

Between one and two percent of all tweets tend to contain geographic co-ordinates. This is interesting information to visualize. However, one percent is not a lot. In order to obtain more data, we have used the user location field in the tweet metadata as an additional clue for the location of the user that we merge with the geographical co-ordinates. If the user location matches one of the 5,222 place names in The Netherlands or Belgium[4] then we use a central point in that place

4. Dutch and Belgian place names were obtained from http://www.d-centralize.nl/projects/6pp/downloads/ (cities.csv) and http://www.rdlt.com/postcodes-van-alle-gemeentes-van-belgie-met-gps-coordinaten

as the presumed position of the user. This strategy increases the frequency of tweets with location positions to about 10%. We use maps from `openstreetmaps.org` with heatmap code written by Patrick Wied for visualizing location information.

As a summary of search results, we also present the 50 most salient words in the search results as a word cloud. These are not selected based on their frequency alone but with the t-test (Church et al. 1991), which compares the relative frequency of a word in search results with its overall relative frequency. The words which appear in the search results more often than could be expected are shown in the word cloud. The t-test is explained in more detail in Section 4.2.

One of the visualizations of a search result is an estimate of the age and gender of the users behind the search results. This information is derived from the user profile by a collection of rules. Besides rules that look for obvious patterns like `"digit digit years"` and `"woman"`, several other clues are used such as first names indicating gender (using lists of 161 frequently used male names and 155 female names) and `"high school"`, `"married"` and `"granddad"` indicating age[5]. We only use three broad age categories: under 18 years, $18--25$ years and older than 25 years, so that we can map as many users as possible into demographic groups. Still we derive this information only for about 25% of the users.

The final "visualization" is a list of identity numbers (ids) of tweets and their associated users. We are not allowed to distribute the tweets but only the ids, so we present the id pairs of the first 30 tweets in the search results to the users, associated with links to the original tweets on `twitter.com`. If the users want to inspect the tweets, they can access them via the links, if they are still available on the Twitter website.

Since we expect that some users will want to make their own visualizations, we added the opportunity to download the numbers underlying the visualizations.

4. Case studies

In order to examine if the tweet collection is useful, we have performed three case studies. In the first, we collected relative frequencies of words and related these to a real-life event. In the second case study, we examined words which frequently appear with certain other words. In the third we inspect discussions taking place on Twitter.

4.1 Frequency of the word "griep" (flu)

In January 2013, The Netherlands was struck by a mild flu epidemic. People stayed home to recover and doctors registered an increasing number of flu patients. This was not an unusual event for the winter months but normally the flu strikes several weeks later.

Our hypothesis is that the flu outbreak is also visible on Twitter, by an increased fraction of tweets containing the Dutch word for flu: "griep". Figure 3 contains the relative frequency of the word measured in the period 25 December 2012 to 24 February 2013 based on a search query which required seven minutes on twiqs.nl. We see graph with peaks and valleys indicating when "griep" was used often or not. The graph was based on about 130 million tweets among which about 36,000 mentioned the flu.

The question is if the relative frequencies in the graph correspond to the fluctuations in the actual cases of flu. In order to test this, we compare the data with two external sources. First, we compare the graph with news articles, analogous to Google Trends[6]. In the time frame of the graph, the Dutch quality newspaper NRC published articles related to flu with the following headlines:

- 3 January: Griepgolf treft Nederland (*Flu engulfs The Netherlands*)

5. Apart from first names, we use 16 words indicating gender, and besides numbers we used 30 words indicating age. Most of these words have been selected based on the appearance in profiles of Twitter users.
6. http://trends.google.com

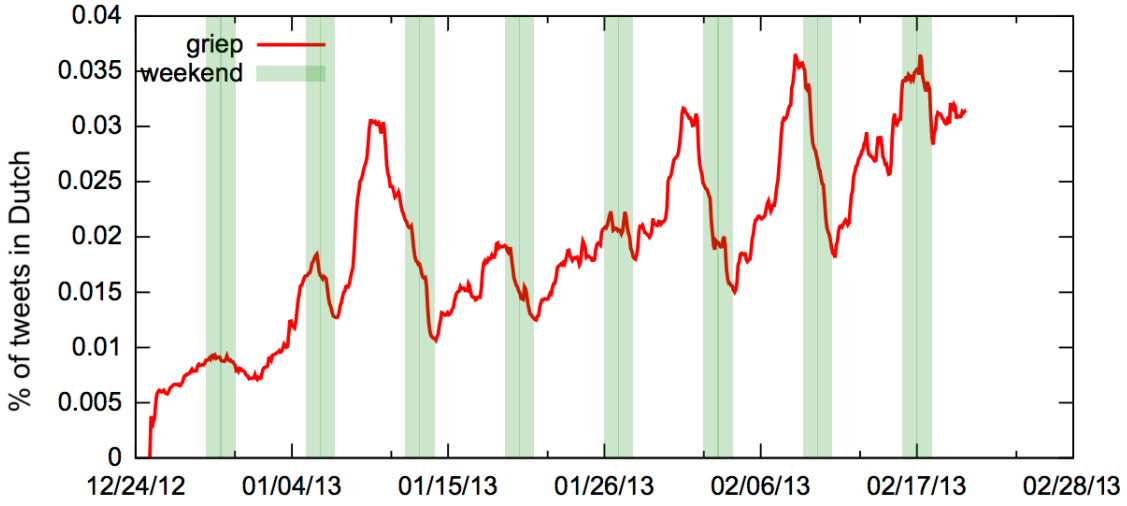

Figure 3: Relative frequency of usage of the word "griep" (flu) in Dutch tweets for the period 25 December 2012 to 24 February 2013.

- 7 January: Eten en vasten in de tijd van griep (*Eating and fasting in a time of flu*)
- 14 January:: Steeds minder mensen blijven thuis met griep (*Fewer people stay home with flu*)
- 17 January: Griepepidemie heeft zich verspreid over hele land (*Flu epidemic has spread over the whole country*)
- 30 January: Griepgolf laait weer op (*Flu on the rise*)
- 21 February: Nog altijd griepepidemie (*Flu epidemic remains*)

Three article titles mention an increase of the flu. The January 3 NRC article speaks of the flu hitting The Netherlands. At that time the word flu was used at Twitter but not frequently. The January 17 article points out that the disease has spread over the whole country. This comes about a week after the first big peak of flu mentions in the tweets. Finally, the January 30 article signals that the flu is coming back. At the publication time of this article, the relative frequency of flu on Twitter was experiencing its second big rise.

Although we find some correspondence between the usage pattern of flu on Twitter and in a newspaper, we have to be careful with marking them as independent effects of real-life events. News media have a broad audience and stories appearing in newspapers are known for having an impact on the discussions taking place on Twitter. Although many tweets report personal experiences, many can also be inspired by events reported in the news.

Another better comparison source of flu information is the website De Grote Griepmeting[7]. Here registered users regularly fill in surveys in which they report on their health, especially with respect to flu-related disease symptoms. For the time period that we are interested in, the flu rate reported by the degrotegriepmeting.nl corresponds to the blue line in Figure 4. We see many correspondences between the frequencies of the flu tweets and the flu rate, for example the shared peaks on 6, 19 and 31 January and 11 and 15 February. But there are also differences: the tweet rate rises in the week

7. `degrotegriepmeting.nl`

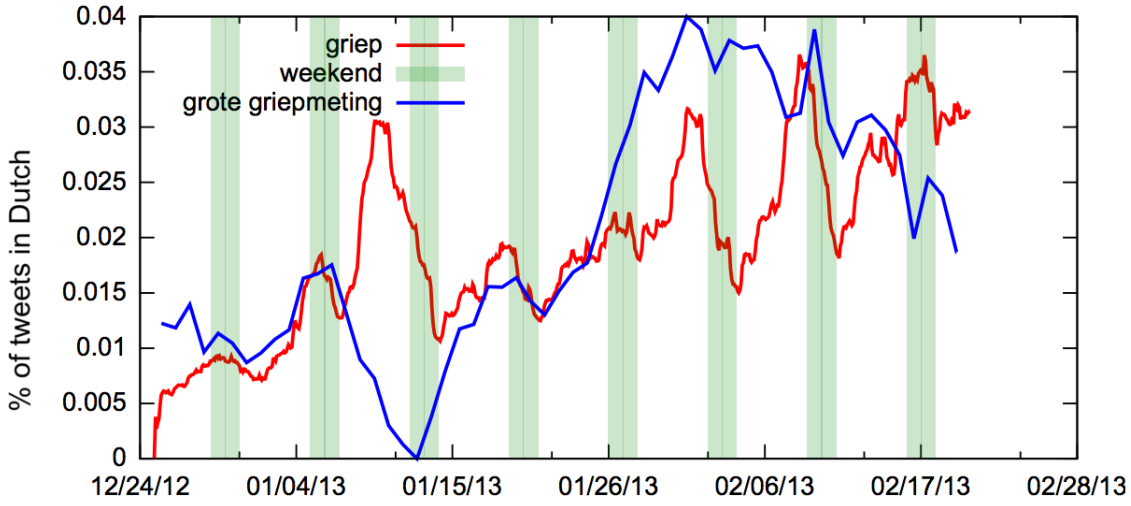

Figure 4: Relative frequency of usage of the word "griep" (flu) in Dutch tweets for the period 25 December 2012 to 24 February 2013 (red line) compared with flu cases in The Netherlands reported from the daily reports of De Grote Griepmeeting (values 400-1200).

of 6 to 12 January while the flu rate is dropping. The same seems to be happening in the weekend of 16 and 17 February. Further study of the tweets is required to reveal what could have caused these differences.

4.2 Which words frequently occur together?

It is interesting to find words that frequently appear in tweets containing a certain keyword. In order to find such words, it is not enough to examine tweets with the keyword and rank the words in these tweets by frequency. The top of such a ranking would be dominated by uninteresting function words like *the*, *a* and *of*. Interesting words can be found by comparing the ranked associated words with a general frequency list. Words that appear high in the first ranking and low in the second, occur more often with the keyword than could have been expected from their general frequency. These words are more interesting to examine.

The task of finding salient words can be performed automatically with the t-test (Church et al. 1991), which compares the probability of a word co-occurring with a keyword, $P(word \mid keyword)$, with the overall probability of the word $P(word)$:

$$t = \frac{P(word \mid keyword) - P(word)}{\sqrt{\sigma^2(P(word \mid keyword) + \sigma^2(P(word))}} \tag{1}$$

$$\text{where } P(w) = \frac{f(w)}{N} \text{ and } \sigma^2(P(w)) \approx \frac{f(w)}{N^2} \tag{2}$$

Here N represents the total number of words and $f(w)$ the number of times that the word w was observed. Like Church et al. (1991), we use add 0.5 smoothing in order to avoid zeroes in the computation.

After searching for a keyword, we count the words appearing in the results, compute t-scores for them by comparing the frequencies with those of reference data (one day of tweets corresponding to the day before the first day in the search period) and rank the words according to the t-scores. As

Figure 5: The top 50 words that occur more often than could be expected from their overall frequency in tweets with the word *griep* (flu) for the period 25 December 2012 to 24 February 2013. Larger font sizes indicate larger positive deviations from the overall frequency. This word cloud contains several words related to *griep* like *bed* (bed), *koorts* (fever) and *ziek* (ill).

Figure 6: The top 50 words that occur more often than could be expected from their overall frequency in tweets with the Brabant dialect word *hedde* (do you have) in February 2013. The cloud contains several related words from the dialect, like *da* (that), *goe* (good) and *ofwa* (or not).

an example, Figure 5 shows the fifty words with the highest t-score in tweets selection of Section 4.1: tweets containing the word *griep* (flu) from 25 December 2012 until 24 February 2013. The words are displayed in a word cloud where the size of the characters of the words is proportional to the t-score of the word. The word cloud contains several words related to *flu*, such as *bed* (bed), *koorts* (fever), and *ziek* (ill).

The words that co-occur with specific keywords may be important for language research and market research alike. Figure 6 shows the word cloud associated with the Brabant dialect word *hedde* (do you have) based on tweets from February 2013 (required search time: five and a half minutes). The cloud contains several related words from the same dialect, like *da* (that), *goe* (good) and *ofwa* (or not). These provide insights in word usage and clues for other relevant searches in this domain. To dialect research, querying Twitter allows to provide empirical answers to the currently open question how dialects are used in new media.

Companies can obtain market research information from the list of words used together with their name. Figure 7 shows the fifty most salient words in tweets containing the company name *NS* (Dutch Railways) in the month February 2013 (required search time five and a half minutes). Their name was used often in discussions involving the traffic schedule (*dienstregeling*), snow (*sneeuw*) and delays (*vertraging*). The word cloud gives a quick impression of the contexts in which the company name was used on Twitter in that particular month.

Figure 7: The top 50 words that occur more often than could be expected from their overall frequency in tweets with the company name "NS" (Dutch Railways) in February 2013.

	level	time	user	tweet
1.		15:15	@1	Vandaag moest ik even aan de Buikhuisenaffaire denken.
2.		15:18	@2	Buikhuisen was geen bankster. RT @1: Vandaag moest ik even aan de Buikhuisenaffaire denken.
3.	>	15:30	@1	@2 I know. Alleen is het akelig als de terechte woede over bankiersgedrag trekken van een hetze krijgt.
4.	>>	15:40	@2	@1 Jazeker. Maar ik zou die hetze niet in de schoenen van @3 schuiven. Don't shoot the messenger.
5.	>>>	15:42	@4	Terecht dat mensen zich boos maken op het systeem, obsceen dat te verkleinen tot één boerenbraadlul @2 @1 @3
6.	>>>>	15:54	@5	@4 @2 @1 @3 ik spreek je wel weer als jij 20 jaar in t bedrijfsleven hebt gewerkt. Niet zo naïef!
7.	>>>>	15:55	@1	@4 @2 @3 Exact. Het was een grote club, politiek gesteund en dikwijls publiek verafgood #gekko
8.	>>>>>	15:57	@6	@1 @4 @2 @3 Het monster moet natuurlijk wel een gezicht hebben, anders is het te moeilijk
9.	>>>	15:46	@7	@2 @1 @3 sure maar deze messenger is wel gewapend en gooit

Table 3: Example of a part of a Twitter conversation involving seven different users and a maximum reply level of five. In the conversation, people discuss the verbal attack on a banker by a Dutch celebrity.

4.3 Examining reactions to tweets

On Twitter it is possible to reply to tweets. When they are logged in, Twitter users have the option to click on the reply button next to each tweet and send a response to the tweet. The response will usually start with @USER, where USER is the name of the user who wrote the tweet that is being replied to. This makes it possible for that user to be notified about the reply to his tweet. The id number of the original tweet is also attached to the metadata of the reply. This makes an automatic selection of chains of replies possible.

Such chains of replies are called conversations. When viewing tweets on twitter.com, one has the option to examine groups of replies by selecting *View Conversation*. The related tweets will then be presented chronologically, either on a separate web page or in a marked section of the current web page.

Our Twitter tool offers a similar view on chains of replies. A user query, one of the format @USER, will generate all tweets of the user USER together with all the replies to his tweets that contain the phrase *@USER*. In the search results, the related tweets will be presented together, with extra characters marking their relation.

Table 3 contains an example, with anonymized users. The reply levels are indicated with > signs. Each message with n signs is a reply to the last message with $n-1$ signs. For instance, message 3 is a reply to message 2 and message 9 is a reply to message 4. Message 2 is special in this conversation. The user selected Retweet (RT) as a reaction to message 1 rather than Reply. Since the Reply link is missing, message 1 is not part of this conversation.

Having access to the conversation structure enables researchers to understand the tweets better. The structures enable conversation analysis, for example for checking if and how positions of the participants change during the conversation. The conversation structure is also interesting for helpdesks that operate via Twitter. They can collect the conversations and check response rates and costumer satisfaction.

Some disclaimers have to be made here. First, as shown by the first two tweets in Table 3, not all related tweets can be identified by the reply relation in the metadata. Second, the tweet collection used by our tool is incomplete, so most conversations retrieved by the tool will be incomplete as well. And third, because of restrictions imposed by Twitter, our tool cannot show the tweet texts to users. An analysis such as the one performed in this section can presently only be performed with the data from twiqs.nl by ourselves.

5. Concluding remarks

In this paper, we described an infrastructure for searching Dutch tweets. We described how we selected tweets, stored them and made them searchable and showed how we visualized search results. Finally we presented three case studies as examples of how the tweet collection can be used.

The first three of the four research questions in the introduction section have been answered. The first question was what useful tweet information can be made available without making the tweets themselves available. We have shown that summaries of tweets in the form of visualizations are interesting for studying several topics (Section 4). The second question was how to deal with the large volume of tweets. We store the data on a parallel architecture which makes it possible to provide search results in reasonable times (Section 3.3 but see also below). The third question concerned the presentation format of search results to the user. We offer the users five different visualizations of the search results and also allow them to download tables with numeric data from which the visualizations were derived (Section 3.5).

This leaves only the fourth question to be answered: is the restricted information we offer valuable for users? The resource is being used daily by different users. Despite the short time of availability (six months) its usage has already been mentioned in several studies: (Kunneman et al. 2013, Liebrecht et al. 2013, van Halteren and op de Weegh 2013, van Wijngaarden et al. 2013).

There are three common issues raised by the users. The first is that they would like to obtain access to the text of the tweets. This is something we cannot offer them because of the usage restrictions of Twitter. This common demand shows there would be interest in a tool that operates differently, collecting tweets from the user's computer and thus lifting the text redistribution constraint[8].

A second common issue raised by the users is coverage. We only manage to collect about 40% of the Dutch tweets because of upper limits on the number of downloaded tweets per time period imposed by Twitter. We cannot satisfy search needs that require complete search results. Users with such a request are referred to the three certified data resellers DataSift, Gnip and Topsy, and to the search facility on `twitter.com/search`. It must be noted that Twitter itself states that they are unable to guarantee that search results are complete.

The final issue concerns the search speed. Although results are computed on a parallel architecture, results do not appear as quickly as users wish. A search query involving years of tweets may

8. Anybody may retrieve historic tweets from twitter.com which are from a specific user or contain specific words. A tool which would support retrieving, storing and analyzing such tweets could allow the user to inspect tweet texts.

take hours to complete. An additional problem is that the parallel Hadoop architecture we use, is shared by different users. This means that on busy hours the available computing time is limited and queries may take a lot longer to complete.

We limit access times by keeping all earlier search results, thus creating a cache with fast response time. However, like Gupta et al. (2013) we can question if Hadoop was the right framework for this task. With respect to storage, the answer would be yes, but with respect to retrieval times the answer would be maybe not. An information retrieval system using multiple index files would operate faster. However, as mentioned earlier in the paper: we do not know how to deal with metadata search in such a set-up.

We have several ideas for future work. These involve technical improvements of the service like offering AND queries, an API interface, a command line interface and login via general organization credentials (Shibboleth). We would also like to test storing the tweets in a multi-index information retrieval system in order to improve access speed. It would also be interesting to perform a user evaluation to get more information on the information needs of users. The feedback we have obtained until now has proven to be very useful.

Acknowledgments

The research described in this paper was made possible by a grant from the Netherlands eScience Center. We wish to thank three anonymous reviewers for valuable comments on earlier versions of this paper.

References

Church, Kenneth, William Gale, Patrick Hanks, and Donald Hindle (1991), Using statistics in lexical analysis, *in* Zernik, Uri, editor, *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*, Lawrence Erlbaum Associates.

Gupta, Pankaj, Ashish Goel, Jimmy Lin, Aneesh Sharma, Dong Wang, and Reza Zadeh (2013), WTF: the Who To Follow service at Twitter, *Proceedings of the 22th International World Wide Web Conference (WWW 2013)*, Rio de Janeiro, Brazil, pp. 505–514.

Kunneman, Florian A., Ali Hürriyetoglu, and Antal van den Bosch (2013), *Supporting open-domain event prediction by using cross-domain Twitter messages*, Talk presented at Computational Linguistics in The Netherlands (CLIN-2013).

Liebrecht, Christine, Florian Kunneman, and Antalvan den Bosch (2013), The perfect solution for detecting sarcasm in tweets #not, *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Association for Computational Linguistics, Atlanta, Georgia, pp. 29–37.

Liu, Xiaohua, Shaodian Zhang, Furu Wei, and Ming Zhou (2011), Recognizing named entities in tweets, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, Portland, Oregon, pp. 359–367.

Petrović, Saša, Miles Osborne, and Victor Lavrenko (2010), The Edinburgh Twitter Corpus, *Computational Linguistics in a World of Social Media*.

Tjong Kim Sang, Erik (2011), Het gebruik van Twitter voor taalkundig onderzoek, *TABU: Bulletin voor Taalwetenschap* **39** (1/2), pp. 62–72. (in Dutch).

Tumasjan, Andranik, Timm Sprenger, Philipp Sandner, and Isabell Welpe (2010), Predicting elections with Twitter: What 140 characters reveal about political sentiment, *Proceedings of the Fourth AAAI conference on Weblogs and Social Media*, pp. 178–185.

Twitter Inc. (2013), *Developer Rules of the Road*, https://dev.twitter.com/terms/api-terms (retrieved May 2013).

van Halteren, Hans and Maarten op de Weegh (2013), *Clues for autism in Dutch tweet production*, Talk presented at Computational Linguistics in The Netherlands (CLIN-2013).

van Wijngaarden, C., C. R. Verschoor, and C. Bonenkamp (2013), *Classifying home locations of Twitter users*, Technical report for the course Advanced Information Retrieval, University of Amsterdam.