Clustering tweets with K-means

Niels Eigenraam
s0620157
c.eigenraam@umail.leidenuniv.nl
LOT 2016 - digital text analysis

Introduction

In Eigenraam (2013), it is suggested that auction websites like Marktplaats.nl can be a valuable resource for dialect studies, especially those aimed at geographical distributions. However, in light of the position of privacy-related matters in today's public discourse the value of Marktplaats and similar websites, as a corpus for linguistic inquiry is questionable. The ethical issues surrounding the use of this kind of data are complicated. It is often the case that data use is restricted by a User Agreements and licensing that prevents the sharing of data with others. Hence the method is not only ethically questionable, but also hard to (re)produce without explicit consent.

Fortunately, the internet is a large and resourceful place. Despite recent alarms regarding the declining number of users, Twitter is still a virtually unbounded source of up-to-date language data. Studies like Goncalves and Sanchez (2014), E. F. Tjong Kim Sang (2011) and E. Tjong Kim Sang and Bosch (2013) prove the usefulness of Twitter data for linguistic research.

In what follows, I will explore the usefulness of tweets and their metadata for linguistic research, with a focus on dialectology.

Twitter

Twitter offers access to their data through several *Application Program Interfaces* (APIs). Developers may use an API to create custom applications

that interface with a website. Relevant to this paper are the *streaming* API (https://stream.twitter.com/1.1/statuses/filter.json) and the *search* API (https://api.twitter.com/1.1/search/tweets.json). Additional documentation for these endpoints can be found at https://dev.twitter.com/streaming/overview and https://dev.twitter.com/rest/public/search.

The search API functions like Twitter's advanced search application. It allows us to search for tweets based on a variety of characteristics such as location, language, and keywords. Unfortunately, when using this endpoint, downloading is restricted to one HTML request per minute (or actually, 15 per 15 minutes). The amount of tweets per request depends on the search query, and is generally quite low. Hence, gathering a decently sized corpus takes quite some time.

The feature most relevant to this paper is searching based on location. Twitter provides a convenient way of collecting tweets based on location in their search API. However, this method relies on users that explicitly add their location to their tweets. Hence, the entries in this metadata field are irregular at best, additionally, in a considerable amount of cases the location is fictional. Both these aspects make the search API somewhat problematic for linguistic analysis.

In addition to the search API, Twitter also offers a streaming service. This endpoint is also limited, but rather than http-requests it limits the number of available tweets to 1% of all tweets. Streamed tweets can include detailed geographical metadata. When users activate the geolocation feature that is built into Twitter, their tweets are tagged with the Cartesian coordinates of their location, as well as other relevant metadata such as place type (city, region, or country), country, and a set of coordinates that form a bounding box around the place. The stream can be filtered on geographical location by specifying a 'bounding box' of coordinates that form a box containing the area of interest. The API then filters out tweets that originate within this box. Crucially, this concerns tweets by users who have switched on Twitter's geolocating feature. Hence the metadata is as accurate as the user's gps allows, and moreover, it is uniformly formatted.

Data

Because of the limitations of the search API I have decided to create a corpus using tweets gathered using the streaming API. Where studies

like Goncalves and Sanchez (2014) and E. Tjong Kim Sang and Bosch (2013) make use of corpora that contain millions of tweets, my own corpus consists of just a fraction of that. This is partly due to time constrains. For instance, the corpus used in Gonçalves & Sànchez was created over the course of two years. Additional time was taken up by writing the software required for downloading tweets and by the creation of a database.

Initially, I planned to look at language variation in Belgium and the Netherlands. Hence, I used the bounding box shown in Figure 1 to filter the tweets. In order to be maximally inclusive, the box includes parts of Germany and France, as well as the French-speaking part of Belgium. The resulting corpus consists of 192 416 tweets. The geographical distribution is shown in Table 1.

Figure 1: Bounding box for the Netherlands and Belgium

Table 1: geographical distribution

country	tweets
Nederland	82 849
België	43 589
France	42 080
Deutschland	22 648
Luxembourg	1 195

country	tweets
R.D. Congo	1
Bangladesh	1
United Kingdom	1
United States	1
Total	192 416

Because of the size of the bounding box, however, a large amount of tweets had to be filtered out. I decided to keep only the tweets produced in Belgium and the Netherlands, which led to the removal of 65 978 tweets (Table 2).

Table 2: distribution BE - NL

country	tweets
Nederland	82 849 (0.66)
België	43 589 (0.34)
Total	126 438 (1.00)

The distribution of tweets over the ten most productive cities is shown in Figure 2. Most striking here is the prominence of Amsterdam and Brussels, although this in maybe not that surprising given both cities' importance.

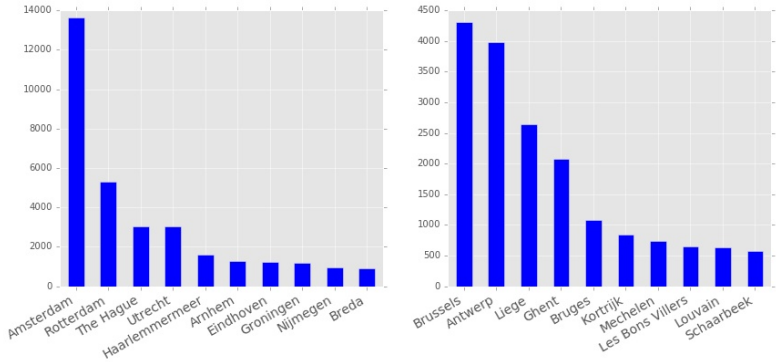

Figure 2: Top 10 most productive cities

As Figure 3 shows, the data in this figure include several languages Dutch. Most important among these are English (32 678 tweets), French (10 130 tweets) and 'und' (9 470 tweets). It is unsurprising that English is this prominent, as it is 'the language of the internet'. The presence of French is unsurprising as well, because Belgium is Dutch-French bilingual. 'Und' stands for 'undefined'. It is the value of the tweet's 'lang' variable when Twitter's language detection algorithm is unable to detect its language (see also https://dev.twitter.com/rest/reference/get/search/tweets).

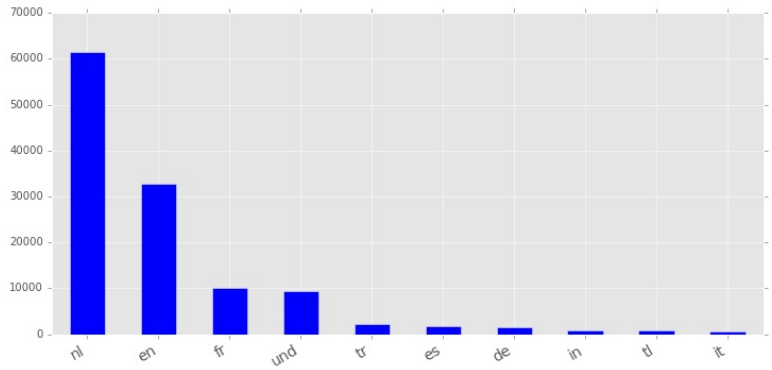

Figure 3: Languages present in the data

Because I am interested in tweets written in Dutch, tweets in the other languages have to be removed. E. Tjong Kim Sang and Bosch (2013) describes two methods that can be used to obtain tweets in Dutch. The first method involves searching for a selection of Dutch keywords and hashtags. The second strategy is to include only tweets that have been tweeted by 5000 of the most productive Twitter-users who are likely to tweet in Dutch. Both methods are not foolproof, hence E. Tjong Kim Sang and Bosch (2013) additionally used a language checker to identify Dutch tweets.

It must be noted that E. Tjong Kim Sang and Bosch (2013) used a Hadoop cluster to process their tweets. Since I do not have a similar infrastructure at my disposal, I will use a different approach in order to eliminate non-Dutch tweets. The metadata accompanying tweets contains two language related variables. The first, simply called "lang", contains the language the tweet is written in as identified by Twitter's algorithm. The second variable ("user_lang") contains the language of the user's interface. Here, I assume that a user who uses the Dutch interface will also predominantly tweet in Dutch.

In order to limit the dataset to tweets written in Dutch, I reduced the data to tweets originating in the Netherlands and Belgium. Additionally, I further reduced it by filtering out all tweets for which "user_language" and "lang" are anything else than Dutch. After these operations we are left with 43 360 tweets Table 3. This is roughly less than 25% of the original data set.

Table 3: distribution of tweets after reduction

country	tweets
Nederland	33 356 (.77)
België	10 004 (.23)
Total	43 360 (1.00)

When comparing the "before and after" distributions of most productive cities, it is interesting to see that Antwerp has overtaken Brussels in productivity, although this might not be as surprising considering that Brussels is the seat of the EU parliament. Another major difference is the fact that Amsterdam is still leading in the Netherlands, but its lead is now

much smaller (Figure 4).

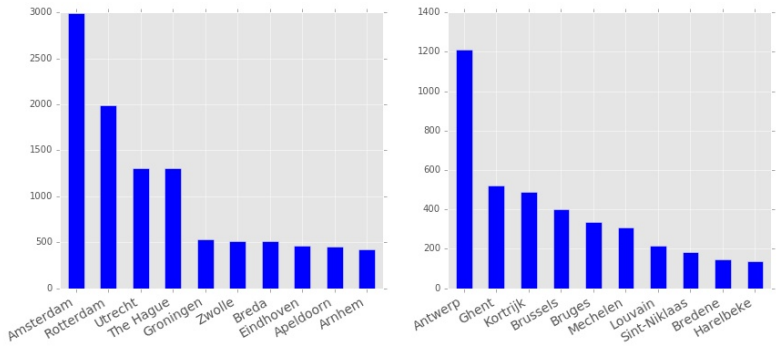

Figure 4: Top 10 most productive cities 2

Analysis

Unfortunately, the dataset that I described in the previous section was too small to find anything interesting. Hence, in what follows I will use a corpus consisting of all tweets present in my data, excluding those that lack the necessary metadata.

For this analysis I am using the 'bag of words' approach. This is a way of analysing text that ignores the syntactic content of the words present in a document. The approach consists of three steps: words are tokenized, the frequency of occurrence of each word is counted, and finally the counts are normalized with regard to de corpus. Using this process, each document in the corpus is transformed into a numerical vector, representing the features present in that document. The corpus as a whole is then represented by a matrix with one row per document and one column per feature (see http://scikit-learn.org/stable/modules/feature_extraction.html).

Initially I treated each tweet as a separate document. However, because tweets are max. 140 characters long, they do not provide the necessary feature information and hence the feature set becomes too large. That is why instead of considering each tweet a separate document I have categorized them according to city. This approach is similar to that of Goncalves and Sanchez (2014), who bin their tweets according to geographical location

(p. 2). They were able to do this because they have vectorized their data using predefined features. In my case, however, the corpus of tweets proved too small to allow this. That is why I have vectorized the corpus using a Term Frequency-Inverse Document Frequency (TFIDF) vectorizer. This vectorizer counts token occurrences for each document (term frequency), and then applies weights to the counts depending on the frequency of the token in the entire corpus: words with a high frequency within a document but a low frequency in the corpus receive a higher weight (inverse document frequency). Applying the vectorizer to the corpus resulted in a matrix consisting of 50 rows and 8799 columns.

Although it is possible to reduce the number of features using Principal Component Analysis (PCA) or Multidimensional Scaling (MDS), I decided not to do this. The high dimensionality may lead to overlap between features, but this is not as much a concern for my investigation as it is for Goncalves and Sanchez (2014). I applied the K-means algorithm to the matrix with K=5. In order to avoid getting stuck in local optima, I ran the algorithm 5 times.

Table 4 shows the five resulting clusters with the ten features that are closest to each cluster's centroid.

Table 4: KMeans clusters

cluster	features	cities
0	antwerpen, gent, leuven, belgium, nmbs, hoplr, campus, at station wel, heb	Ghent, Antwerp, Bruges, Louvain, Kortrijk, Mechelen,
1	cest, des, jai, pour, mais, est, une, tu, qui, sur	Wemmel, Thionville, Schaarbeek, Tourcoing, Villeneuve-d'Ascq, Roubaix, Sint-Gillis, Dunkirk, Amiens, Brussels, Lille, Liege

cluster	features	cities
2	ich, und, hpa, ist, nicht, auf, auch, von, zu, kmh	Aachen, Duisburg, Menameradiel, Düsseldorf, Dortmund Essen, Bonn, Cologne
3	amsterdam, schiphol, airport, in amsterdam, rit, netherlands	Amsterdam, Haarlemmermeer
4	wel, weer, heb, meer, goed, kca, uit, deze, moet, votemainstreet	Almere, Nijmegen, Tilburg, Spijkenisse, Maastricht, Dordrecht, Rotterdam, Amersfoort, Groningen, Arnhem, Eindhoven, 's-Hertogenbosch, Utrecht, The Hague, Zwolle, Zaanstad, Enschede, Leyden, Deventer, Haarlem, Apeldoorn, Breda

The clusters are not that surprising. German and French tweets are neatly separated from the Dutch tweets, as could be expected. More interesting is the separation between tweets from the Netherlands (cluster 4) and those from Flanders (cluster 0). It is also somewhat amusing to see that Amsterdam has a cluster on its own, although if we look at the centroid words it is not so big a surprise.

In addition to the K-mean clustering, I have also calculated the Cosine Distances for each document in the corpus. Each row in the resulting distance matrix consists of a vector containing distances between that document and the other documents in the corpus. Subsequently I used MDS to reduce the distance matrix to a two-dimensional one. Figure 5 shows the resulting scatter plot.

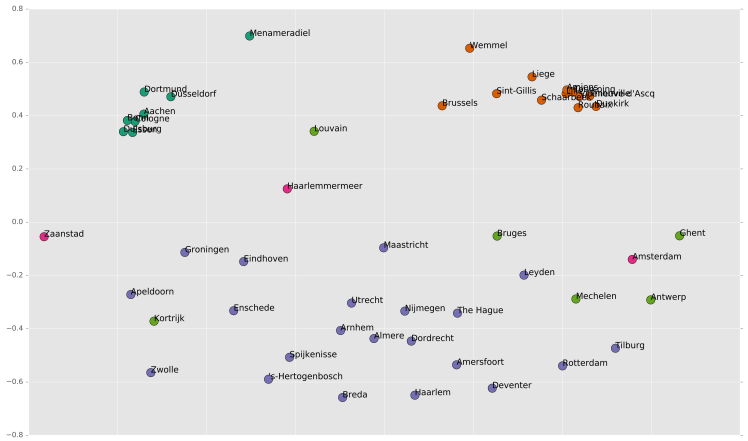

Figure 5: Tweet clusters

Discussion and conclusion

Again this shows the neat clustering of the French and German tweets. However, interesting enough in Figure 5 the tweets from the Belgian city Kortrijk are closer to the Dutch cities Deventer and Spijkenisse than they are to any other city. The same goes for tweets from Amsterdam, Rotterdam and Menameradiel (a municipality in Friesland). The most extreme case, however, is the distance between Zaanstad, Haarlemmermeer and Amsterdam. Although geographically, these cities are very close, based on the content of their tweets they are very far apart.

The goal of this paper was to use clustering algorithms to identify meaningful differences between 'kinds' of Dutch. Unfortunately, Table 4 shows that none were present in this corpus. I attribute this to the size of my corpus. I expected twitter to be a kind of unlimited resource for linguistic data – and perhaps it really is –, however, gathering a sufficient amount of tweets to build a corpus proved to be harder than I thought. Additionally, the actual data is a lot less rich than I expected it to be, mainly because of the 140 character limit.

References

Eigenraam, C. (2013). Een Hengel Voor Te Vissen: Marktplaats.nl Als Corpus Voor Onderzoek Naar Regionale Variatie. *Nederlandse Taalkunde*, *18*(2), 215–221. http://doi.org/http://dx.doi.org/10.5117/NEDTAA2013.2.EIGE

Goncalves, B., and Sanchez, D. (2014). Crowdsourcing Dialect Characterization Through Twitter. *PLoS ONE*, *9*(11). http://doi.org/http://dx.doi.org/10.1371/journal.pone.0112074

Tjong Kim Sang, E. F. (2011). Het Gebruik van Twitter Voor Taalkundig Onderzoek. *TABU*, *39*(1-2), 62–72.

Tjong Kim Sang, E., and Bosch, A. van den. (2013). Dealing with Big Data: The Case of Twitter. *Computational Linguistics in the Netherlands*, *3*, 121–134.