Contents

1 Introduction (5p) 2

1.1 Twitter . . . 2

1.2 General Idea . . . 2

1.3 Regional word attempt . . . 3

1.4 Geo location attempt . . . 3

1.5 Expectations . . . 3

1.6 Sources, used corpora . . . 3

1.7 Region map . . . 3

2 Algorithms 3

2.1 Cosine similarity . . . 3

3 Regional word attempt (8p) 4

3.1 Detailled description of the idea . . . 4

3.2 Source of the data and creation of the CSV . . . 4

3.3 Why the loops are so important . . . 4

3.4 Experiments . . . 4

3.4.1 Parameters . . . 4

3.4.2 Expectations . . . 4

3.4.3 Discussion . . . 4

3.4.4 -> new Experiment . . . 4

3.5 Conclusion . . . 4

4 Geo location attempt 4

4.1 Source of the data . . . 7

4.2 Why the loops are so important . . . 7

4.3 Experiments . . . 7

4.3.1 Parameters . . . 7

4.3.2 Expectations . . . 7

4.3.3 Discussion . . . 7

4.3.4 -> new Experiment . . . 7

4.4 Conclusion . . . 7

5 Conclusion (3p) 7

5.1 Results of both attempts compared . . . 7

5.2 Reasons for good/bad results . . . 7

5.3 Outlook: What could be better if we had enought time . . . 7

1 Introduction (5p)

1.1 Twitter

Twitter ist ein seit 2006 bestehender und heute weltweit genutzter Webdienst zur Versendung von Kurznachrichten (sogenanntes Mikroblogging). Nur ein kleiner Teil der Nachrichten, der sogenannten Tweets, ist in deutscher Sprache verfasst. Die überwiegend genutzten Sprachen sind Englisch, Spanisch, (...). Auch im relativen Vergleich zur Einwohnerzahl zeigt sich, dass in Deutschland, Österreich und der Schweiz wenig getwittert wird, während das Netzwerk etwa in den Niederlanden, Großbritannien, Japan und Indonesien extrem populär ist.

Auch innerhalb Deutschlands ist die räumliche Verteilung versendeter Tweets auf Grundlage mitgesendeter Geodaten ermittelbar. Sie spiegelt größtenteils die Bevölkerungsverteilung wider; Zentren sind vor allem Berlin und das Ruhrgebiet. Allerdings ist - neben dem Twittern allgemein - im Besonderen das Mitsenden von Geodaten (sogenanntes Geotagging) im deutschsprachigen Raum eher unpopulär.

1.2 General Idea

Ziel: ungefähre regionale Einordnung eines Tweets innerhalb des deutschssprachigen Raums trotz der o. g. seltenen Geodaten und schlecht benutzbaren Herkunftsangaben Idee: Sprache verrät Herkunft, also sollte aus dem reinen Tweettext die Herkunft ablesbar sein

Es existieren hier zwei unterschiedliche Größen. Zum einen gibt es den oder die Orte, an denen ein Twitterer aufgewachsen ist und die ihn sprachlich geprägt haben. Hauptsächlich diese Orte messen wir, wenn wir nach mundartlichen Ausdrücken und Ausdrücken der regionalen Alltagssprache suchen. Auf der anderen Seite steht der momentane, mitunter sehr kurzfristige Aufenthaltsort, von dem aus der Nutzer twittert. Auf Inhaltsebene der Tweets wird er sich eher in ortsbezogenen Begriffen (Ortsnamen, Verkehrsknotenpunkte, Lokalitäten, lokale Ereignisse und Persönlichkeiten etc.) wiederspiegeln. Er ist es außerdem, den wir aus den Geodaten von Tweets erfahren. Nun sind Geodaten jedoch die einzigen Daten, die wir zur Evaluierung unserer Ergebnisse verwenden können. Während unser geodatengestützter Ansatz damit recht passend evaluiert werden kann, zielt unser Regiowort-Ansatz speziell auf die Größe 'sprachliche Herkunft des Twitterers' ab und wird die Evaluierung daher zwangsläufig mit einem gewissen Handicap absolvieren.

Ansatz: - Einteilung des deutschsprachigen Raums in Regionen - Machine Learning auf Trainingsdaten aus diesen Regionen - Bag-of-words model (Betrachtung von Unigrammen)

$$sim(\vec{q}, \vec{d_j}) = \frac{\sum_{i=1}^{N} w_{i,q} \times w_{i,j}}{\sqrt{\sum_{i=1}^{N} w_{i,q}^2} \times \sqrt{\sum_{i=1}^{N} w_{i,j}^2}}$$

Figure 1: Cosine similarity

$$\begin{aligned} q &= \text{'Ich glaube @Drahflow tippt noch schneller als er redet. ;) \#om13'} \\ \vec{q} &= (0.427, 0.38, 0.4, 0.39, 0.391, 0.602, 0.41) \\ \vec{\vec{d}} &= \frac{\sum_{i=0}^{N} d_i}{N} = (0.376, 0.336, 0.349, 0.346, 0.361, 0.481, 0.379) \\ sim(\vec{q}, \vec{\vec{d}}) &= 0.9986 \end{aligned}$$

Figure 2: Example for the similarity calculation

1.3 Regional word attempt

1.4 Geo location attempt

1.5 Expectations

1.6 Sources, used corpora

1.7 Region map

2 Algorithms

2.1 Cosine similarity

Although Tweets may differ from one region to another in some way, a huge percentage of the German Twitter users write their messages exclusively in standard German with no signs of any regional influence whatsoever.
For example the following Tweet just appeared in my timeline:

> 'Ich glaube @Drahflow tippt noch schneller als er redet. ;) #om13'

Keeping in mind that usernames, hashtags, smileys and punctation are being removed in the pre-processing, this Tweet contains way to ordinary words to be assigned to a specific region. It is written in pure High German and therefore could be sent from a town in Bavaria as well as from Berlin and depending on the data we use to train our algorithm with, the result of the program could be 'Austria' or 'Northern Germany' as well.

To face this problem we decided to filter this kind of undistinguished Tweets to have more reliable results and because of that also increase the accuracy of the algorithm.
Our idea was to determine the average Tweet-vector $\vec{\vec{d}}$ of all the Tweets in the training-corpus and to compare the Tweet-vector $\vec{q}$ of a given Tweet with it, using the cosine

metric (see Figure 1 on page 3) as recommended in Jurafsky & Martin. If the similarity between both vectors is smaller than a specified threshold, we continue to map the Tweet to one of the region, if not we stop and return a message, that the Tweet is written in standard German and can not be classified. In the example in figure 2 on page 3, the vector $\vec{q}$ for the Tweet mentioned above is compared to the average Tweet-vector $\vec{\vec{d}}$, returning a very high similarity of 0.9986. Nevertheless the algorithm states, that the Tweet was most likely sent from Switzerland.

But the most difficult thing was to find the right threshold, that separates the too ordinary Tweets from the regional ones. To make a guess which percentage of all Tweets are written in standard German, we calculated the cosine similarity of all Tweets in the training-corpus to the average vector, sorted them and had a look at the distribution (figure 3 in page 5).
The result was not surprising: More than 80% of all Tweets had a similarity of 0.9 or higher. Or, seen from another point of view, only 20% of all Tweets differ enough from the average vector to calculate reliable results.

In the experiments in the sections 3 (regional based attempt) and 4 (geo location based) we tested different thresholds to find a good balance between a reliable classification with a high accuracy and the coverage of as many Tweets as possible.

3 Regional word attempt (8p)

3.1 Detailled description of the idea

3.2 Source of the data and creation of the CSV

3.3 Why the loops are so important

3.4 Experiments

3.4.1 Parameters

3.4.2 Expectations

3.4.3 Discussion

3.4.4 -> new Experiment

3.5 Conclusion

4 Geo location attempt

As stated in the previous chapter, the foundation of all calculations in the regional word attempt is a list of a few hundred words that appear more likely in a specific region. Although this list and their probability distribution is based on scientific research it markes the weak spot of this attempt for number of reasons. For example people in a specific region use a typically word in their everyday language while speaking to their friends or family, but there is no proof that this people also use this words in their written

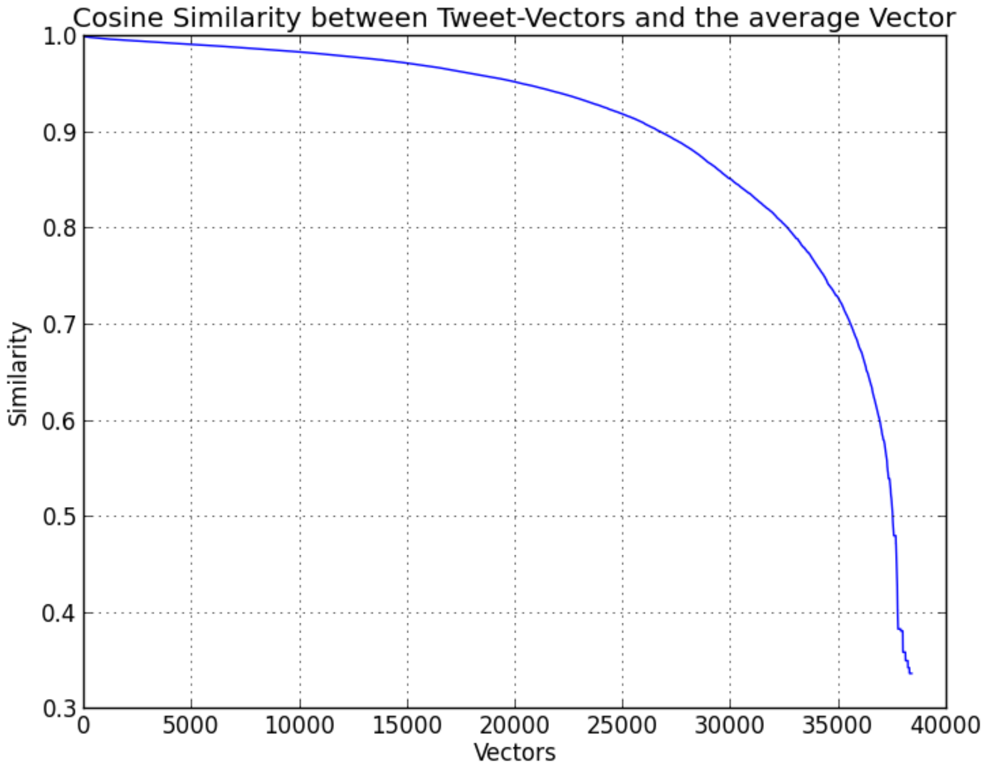

Figure 3: Cosine similarity between all Tweet-vectors and the average Tweet-vector

language, even if it is only their private Twitter account. In addition, the list is way to short to cover only fraction of the words, people use on Twitter, so the end results mostly rely on the data that is generated in main loop of the algorithm.

We seem to have no other choice but to trust this generated values, so we had the idea of skipping the manually created word list and use an automatically created based on a corpus of Tweets with a geo location instead.
Before entering the main loop, we had to write another algorithm that learns the distribution on all seven regions for all the words in the corpus. This way we are generating a list of words that covers nearly all the words.

In order to classify a tweet, that has a geo location, we had struggled to find a good way to represent the seven regions in a way, that we could easily check from which of them a Tweet was sent. After a few unsuccessful approaches, we decided to define polygons for the regions that are not overlapping each other, but also leave no gaps between them. For the value of the points we simply used their longitude and latitude coordinates, that can be represented as floats. We did not implement a point-in-polygon algorithm ourself, but used the version found here [SOURCE] instead. To find out from which region a tweet was sent, we iterate over all regions and return the first one, where the point-in-polygon function returns true.

4.1 Source of the data

4.2 Why the loops are so important

4.3 Experiments

4.3.1 Parameters

4.3.2 Expectations

4.3.3 Discussion

4.3.4 -> new Experiment

4.4 Conclusion

5 Conclusion (3p)

5.1 Results of both attempts compared

5.2 Reasons for good/bad results

5.3 Outlook: What could be better if we had enought time