

A TWITTER BASED BREXIT ANALYSIS

J. KUNATH

Department of Physics, ETH Zurich

J. GACON

Department of Mathematics, ETH Zurich

DR. I. MOISE

Department of Humanities, Social and Political Sciences, ETH Zurich

AND

DR. E. POURNARNAS

Department of Humanities, Social and Political Sciences, ETH Zurich

May 29, 2017

ABSTRACT

Recently, the politics in Europe are subject to changes that are heavily biased towards the right wing and conservatives. Scenarios which we deem unthinkable occur surprisingly often, the most prominent examples being the so-called “Brexit” and the Trump administration. They all share far-reaching similarities: A dedicated political group pursuing a controversial goal as well as an initially low attention of the opposition. Voters that feel left behind and forgotten see their opportunity to have an impact - the election divides the country.

We are interested in how the attention for such political events changes over time. We chose the Brexit for our case study and the activity on related topics on Twitter as a measure of attention.

1. INTRODUCTION

Britain’s referendum about staying in the EU or leaving it (Brexit for short) has drawn a lot of attention lately. The negotiations for Britain’s exit have begun and the first summit took place in late April. At the same time the presidential elections in France are provoking discussions concerning the EU and its constituents. This attention enables us to investigate the recent Twitter activity on Brexit topics very well. We were crawling data using the Twitter APIs, which give us real-time Tweet access, for several time windows between April 5th and May 10th 2017.

Also, we had access to Twitter archives from the Computational Social Science group at ETH, which allowed us to get Tweets from February, April and May 2016.

We used a set of Brexit-related keywords to filter for relevant input. With the `user_time_zone` as location of the Tweet we can restrict the search to certain regions. As a tool for analysing the useful sources we used Python’s VADER^{1,2} sentiment analysis and our own sentiment analysis, based on the Brexit gold standard³ — a 2000 Tweets database whose sentiments have been assigned manually by an expert group.

Our goal was to investigate the activity of Tweets in favour of and against the Brexit, identify peaks and interesting behaviours and map those to political events.

In 2 we explain our sentiment analysis approach in detail. We continue in 3 by looking at the frequency of keywords over time in different regions. Specifications about the implementations can be found in 4 and finally we present the results in 5.

2. SENTIMENT ANALYSIS

kunathj@phys.ethz.ch
jgacon@phys.ethz.ch
izabel.moise@gess.ethz.ch
evangelos.pournarnas@gess.ethz.ch

A sentiment is an attitude, emotion or opinion towards a topic. As a subjective impression, its assignment is a matter of interpretation. In general, this is a binary decision with two opposing categories: for/against, good/bad. A neutral attitude towards the topic can be added as a third category. As not even humans necessarily agree on the classification of a statement, a certain amount of errors is unpreventable.

Sentiment analysis enables us to learn about other people’s opinions and is hence also called opinion mining. It is essential to a wide range of applications from collecting customers’ attitudes towards a company’s products to social research.

The complexity of human language challenges sentiment analysis. Negations and different tones, especially sarcasm and irony, have to be spotted, slang has to be understood. These problems are approached in natural language processing.

2.1. Keyword-mapping

We chose a word-based analysis approach. In order to improve the outcome, we introduce a preceding filtering: Tweets containing specific keywords are classified immediately. Then we apply a sentiment analysis on the remaining Tweets.

Our training set, the Brexit gold standard, contains 1742 Tweets that are still accessible. The rest has been deleted. The five types are `leave`, `stay`, `no sentiment/don’t care`, `irrelevant` and `undecided`. This paper combines the latter three to `other` and thus only considers three types in total.

The distribution of our tested keywords is shown in fig. 1. Some keywords have a high probability of being used in a Tweet with a certain sentiment.

Our goal now is to develop an algorithm that finds strong keywords which we can use to categorise Tweets. As we know the sentiments of the Tweets in our training

set, we can investigate how often a Tweet containing a specific keyword is in favour of the Brexit, against or neither. Clearly, a Tweet can contain multiple keywords.

Therefore we introduced a round-based selection algorithm:

1. For the training set, find the sentiment distribution per keyword.
2. Select the keywords that are strongly biased towards a sentiment.
3. These words are now classifiers: all Tweets containing a classifier of a category (leave/other/stay) are mapped to this sentiment.
4. Remove all Tweets containing classifiers from the training set.
5. Repeat with the remaining set, until no clear associations can be found anymore.

This process is visualised in fig. 6. After running this algorithm, the keywords with a strong enough tendency towards a sentiment are:

| run | keywords | maps to |
|-----|---|---------|
| 1 | 'ukip', 'no2eu', 'britainout', 'voteleave', 'leaveeu' | leave |
| 2 | 'euref', 'eureferendum', 'takecontrol' | other |
| 3 | '#strongerin', 'remain', 'ukineu' | stay |

We are able to categorize 30% (516) of the available Tweets this way. The achieved mapping accuracy on the training set is 64% for the three-sentiments decision.

2.2. Comparison of sentiment analysers

Although VADER is a strong sentiment analysing tool, it is not applicable to our problem. It tells us if a Tweet contains a positive, negative or rather neutral text. But this does not tell us whether this Tweet supports Brexit or not. The Tweet:

Is losing the City of London really a price worth paying for Brexit? - The Telegraph - <https://t.co/CxGVcfFnk3>

yields a VADER score of -0.108, whereas the following Tweet's score is 0.258:

Working together with our allies is never a mistake.
#wise_choice #Remain #StrongerIN
<https://t.co/CFNHdjNqDD>.

While both Tweets clearly favour remaining in the EU, VADER assigns opposing values to them. A quantitative argument against using VADER can be made with fig. 7 and tables 1 to 3. No clear correlation between Brexit opinion and VADER score can be found.

Our own sentiment analyser rates the two Tweets from above with 0.0230 and 0.0114 and thus correctly labels both of them as **stay**.

| method | distribution (in %) | | |
|--------------|---------------------|---------------|---------------|
| | leave | stay | other |
| training set | 701 40.2 % | 376 21.6 % | 665 38.2 % |
| ssix | 658 37.8 % | 458 26.3 % | 626 35.9 % |
| VADER | 523 30.0 % | 602 34.6 % | 617 35.4 % |
| keywords | 265 51.1 % | 100 19.3 % | 153 29.5 % |
| ssix+keyw. | 668 38.3 % | 500 28.7 % | 574 32.9 % |

TABLE 1

Distributions of the categories for different methods.

Note: The percentages have been rounded, therefore the rows don't necessarily sum up to 100%.

| method | wrong prediction | | | | applications |
|------------|------------------|------|-------|----------|--------------|
| | leave | stay | other | Σ | |
| ssix | 302 | 214 | 359 | 875 | 1742 |
| VADER | 465 | 246 | 350 | 1061 | 1742 |
| keywords | 55 | 31 | 62 | 148 | 516 |
| ssix+keyw. | 294 | 180 | 366 | 840 | 1742 |

TABLE 2

Amount of falsely assigned Tweets for the different methods and sentiments.

We trained it on the Brexit gold standard Tweets. A value which depends on the categorising purity and the occurrence frequency is assigned to each single word. Applying the sentiment analysis on our training set gives us a spectrum that can be approximated with three Gaussians.

$$G : \mathbb{R} \rightarrow \mathbb{R}_{>0}$$

$$x \mapsto c \exp \left(-\frac{(x - \mu)^2}{2\sigma^2} \right), \quad c, \mu, \sigma \in \mathbb{R} \quad (1)$$

We set the discriminants* to the values where two neighbouring (non-normalised) Gaussian functions intersect (fig. 4).

The most important features for our analysis are a category distribution that is close to the correct distribution and a high accuracy of the category mapping. As shown in table 1, both ssix and ssix combined with the keyword-mapping have a Tweet distribution that is close to the actual training set's distribution.

We obtain the best accuracy with the keyword-mapping. But as this method is only applicable to 30% of the Tweets, it has to be combined with another method to allocate the remaining ones. The combined ssix and keywords-method yields the best accuracy for the whole training set. This can be read from table 3 as 51.8%.

3. FREQUENCY ANALYSIS

* The discriminant is the threshold where we decide which category a Tweet belongs to.

| method | right prediction | | | | accuracy |
|------------|------------------|------|-------|----------|----------|
| | leave | stay | other | Σ | |
| ssix | 399 | 162 | 306 | 867 | 49.8 % |
| VADER | 236 | 130 | 315 | 681 | 39.1 % |
| keywords | 199 | 71 | 100 | 370 | 71.4 % |
| ssix+keyw. | 407 | 196 | 299 | 902 | 51.8 % |

TABLE 3

Amount of correctly assigned Tweets for the different methods and sentiments. The keywords method has the best mapping accuracy but works only for a subset of the training data.

| user_time_zone | counts | lang = en |
|----------------------------|--------|-----------|
| London | 253277 | 250033 |
| Pacific Time (US & Canada) | 153069 | 92007 |
| Brasilia | 53457 | 5060 |
| Amsterdam | 38102 | 29641 |
| : | : | : |
| : | : | : |
| Dublin | 14191 | 13304 |
| Paris | 14002 | 4503 |
| Edinburgh | 13107 | 13066 |

TABLE 4

Amount of Tweets sorted by timezone for selected zones.

In order to see how the attention of the Brexit evolves over time we investigated how often certain keywords appeared. The timestamp is available in every Tweet and is easily accessible. For the location the tags **geo** of **place** are provided. Unfortunately by far most of the Tweets have their location disabled and these tags are not available. However, the **user_time_zone** is contained in almost all our data, which does the job in our case. The timezone does not indicate the UTC±HH:MM only, but actually tells you the nearest example city of this time zone. A typical dataset would read:

| created_at | geo | user_time_zone |
|---------------------|------|----------------|
| 2017-04-02 23:19:42 | None | None |
| 2017-04-02 23:19:45 | None | Karachi |
| 2017-04-02 23:19:46 | None | London |
| 2017-04-02 23:19:52 | None | Moscow |

From a test set of 1067045 Tweets, 17909 (1.7%) had a **place** tag, 631172 (59%) a **user_time_zone** and 0 a geolocation. The total number of Tweets for various important timezones are shown in 4.

Now one simply needs to filter the Tweets according to the desired timezones and keywords, group them by day, and count. Results using the timezone filter London and multiple keywords are shown in fig. 2 and the total activity in fig. 3.

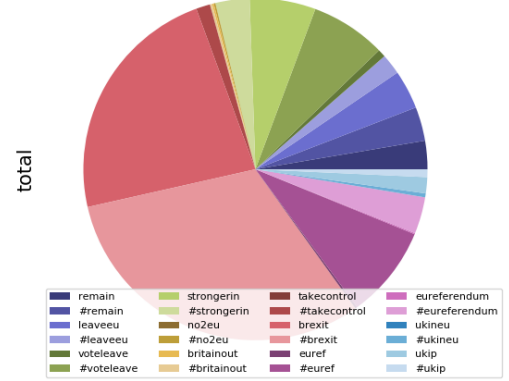


FIG. 1.— Distribution of our set of keywords in the training set. Tweets containing a keyword with # are only counted for the #keyword piece of this pie chart and not again for the keyword without #. The distributions per sentiments can be found in the fig. 6.

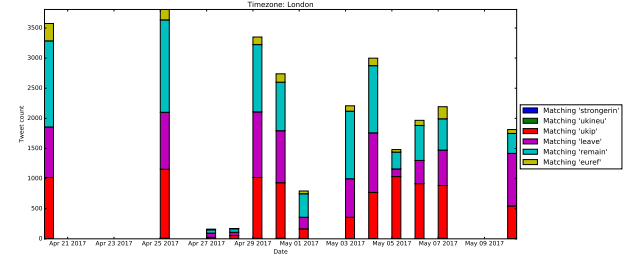


FIG. 2.— We counted the appearances of certain keywords in the Tweets for each day in our data. Words like “ukip” are very present over the whole time window. We excluded the keyword “brexit”, as it is included in almost every Tweet. See fig. 9 for a larger version of the same plot.

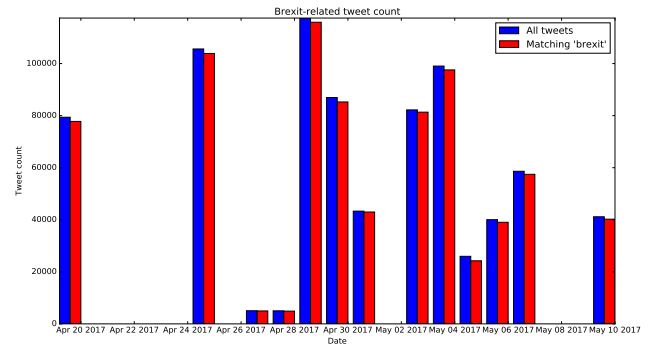


FIG. 3.— Total activity of Brexit-related Tweets. As one can see almost all Tweets contain the keyword “brexit”, thus it cannot be used to classify sentiments. However it is very useful to roughly filter for Tweets about the referendum.

4. IMPLEMENTATIONS

4.1. Language specifications

We chose Python as our working language for several reasons.

Tweepy⁴, a Python library, grants easy access to the APIs. The sentiment analyser VADER is accessible as a Python library via github or the `nlTK` module. Pandas dataframes allow convenient manipulations of the data and Python’s matplotlib can be used to visualise our results.

The gathered Twitter data was stored in a MySQL database.

4.2. Data collection

In order to get access to Twitter’s APIs we created a Twitter account and used our user’s access tokens for identification.

Twitter has four API classes — two of which give us the Tweets, namely the REST APIs⁵ and the Streaming APIs⁶. Latter gives low latency access to the global stream of Tweet data. It is suitable for long-term data mining, as it can simply be kept running in the background. The REST APIs work on a request-response basis, meaning one has to send a request for certain keywords in a certain time and Twitter responds with the data.

For our use the REST APIs had a distinct advantage: one has access to the data of the last seven days, and not only real-time. As we had to crawl our data in a relatively short time we chose this access method over Streaming. We had a Python script running on our server which sends requests for our keyword set on a regular basis and saves the formatted data to the MySQL database. Our keywords are based on the hashtags proposed by Llewellyn and Cram⁷:

```
brexit, strongerin, euref,
eureferendum, leaveeu, ukineu,
voteleave, no2eu, 43brokenpromises,
britainout
```

We tried working with a larger set of keywords containing e.g. `#go`, however the contribution of Tweets chosen by these new criteria turned out rather unsuitable. While we were gathering data of all languages - the hashtags used should stay the same in each language and people may Tweet in english anyway - we only used Tweets of an `'en'` language. Checking foreign language Tweets in our database revealed that they are mainly off-topic and not useful at all. Also, our analyser only works for english words, so a french Tweet with the keyword `brexit` doesn’t help us, as we cannot classify its sentiment.

Drawbacks of the REST APIs are the rate-limiting and the time-outs. We’re only allowed to request a certain amount of data in a certain time, otherwise the interface throws an error. Our script tried to take care of potential limits however sometimes the API threw an unexpected error and our program crashed. We didn’t always notice this immediately, which accounts for some days lacking in our plots. In total we had about 1.3 million Tweets, 913 000 of which were marked as english. 60-64% of all our data were retweets: 767 000 in total and 588 000 in english.

As mentioned in the introduction we were provided with Tweets from 2016 by the COSS group at ETH. They were mining for a slightly different set of keywords. In fig. 8 we see the total count of Tweets per day and the count of Tweets containing the keyword `brexit`. If the two bars are of similar height we expect almost all of Tweets to be relevant. In the data set from 2016 this is not the case and it could be that not all crawled Tweets are important. This, however, is not a severe problem as our sentiment analysis should categorise off-topic Tweets as “other”.

5. RESULTS AND DISCUSSION

Going back to fig. 8 we immediately notice two things: Large fluctuations aswell as a peak at the 29th of April 2017. Days with apparently no Tweets originate from lacking measurements from us. The Tweet counts from 27th and 28th of April are suspiciously low and are most likely due to faulty or interrupted data mining. On the 29th a Brexit-summit took place in Brussels[†] which caused a lot of attention. Some other dates on which we would have expected a rise in attention were: February 20th and 21st, when prime minister James Cameron announced the date of the referendum[‡] and the day later former London major, Boris Johnson, joined the “leave”-campaign[§] aswell as April 15th, when the referendum campaign kicked off[¶]. However in February the plots show no significant increase.

In figs. 9 and 10 we broke the Tweet counts down to some of their constituents. The plot on the top shows our data from late April to beginning of May 2017 and the second row from February, April and May 2016. We observe that the keyword `ukineu` vanishes very quickly and `ukip` rises in popularity. In February 2016 keywords leaning towards “leave” are less prominent than in the following months. In the bottom row we have again the data from 2017 but for different timezones. The overall count distributions are similar for all of the four zones - with two main differences: On April 25th the amount of Tweets in London and the US is significantly above the average. On the contrary, in Edinburgh the count on this day drops by 50%. In the US and Edinburgh we also discover a much lower prevalence of the keyword `leave`. This result fits our expectations as the Scottish people favour staying in the EU^{**}.

Figures 11 and 12 present the results from our sentiment analysis. We immediately notice that an overwhelming majority of the data has been classified “stay”. The category “other” accounts only for a vanishing percentage of all Tweets. In the data from 2016 the composition of the categories does not change decidedly, whereas in 2017 we observe inconsistencies. The Brexit vote took place on June 23rd 2016. We expected the fraction of “stay”- and “leave”-Tweets to change after this date, as the vote was very controversial and the result came as a

[†] <https://www.theguardian.com/politics/2017/apr/29/eu-leaders-set-to-take-tough-stance-in-special-brexit-summit>

[‡] <https://www.theguardian.com/politics/2016/feb/20/cameron-set-to-name-eu-referendum-date-after-cabinet-meeting>

[§] <https://www.theguardian.com/politics/2016/feb/21/boris-johnson-eu-referendum-campaign-for-brexit-david-cameron>

[¶] <https://www.theguardian.com/world/2016/apr/15/european-referendum-campaign-boris-johnson-alistair-darling>

^{**} <http://www.bbc.com/news/uk-politics-36616028>

surprise to many people.

A large amount of young people — the demographic part of the population using Twitter more than any other and in favour of staying in the EU — didn't cast their vote**. Therefore a rise in “stay”-Tweets would have been sensible in the end of June. Unfortunately we have no access to data from this time and if we compare our pre-vote data with the post-vote data we cannot find a significant change.

APPENDIX

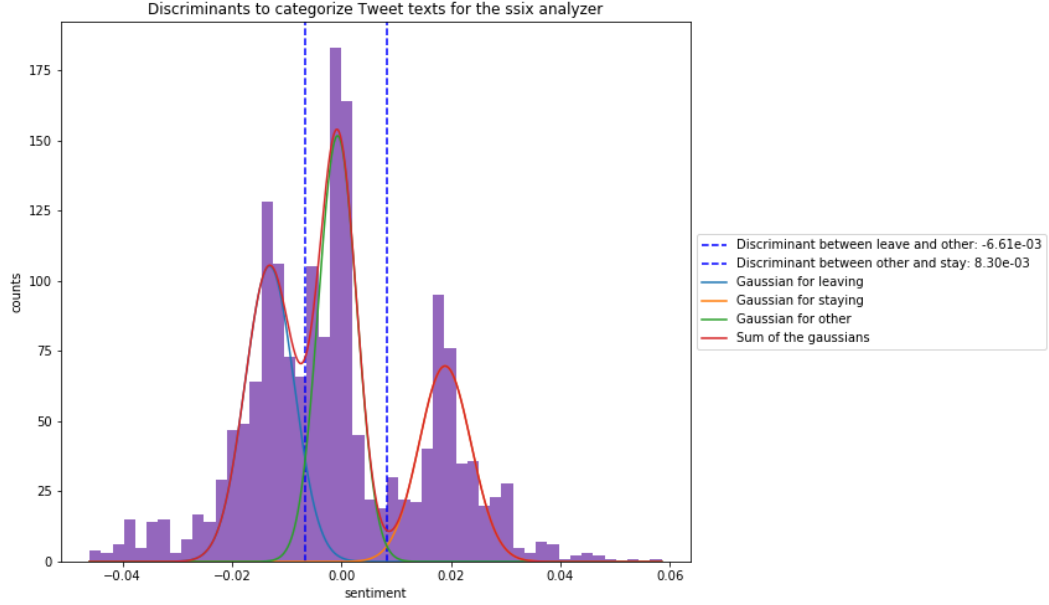


FIG. 4.— Histogram of the sentiment values assigned by our ssix analyzer. The spectrum can be approximated as the sum of three Gaussian distributions. The discriminants are set to the values where the Gaussians intersect. The three functions are $G(\mu, \sigma, \text{norm}) = G(-0.0131, 4.47 \cdot 10^{-3}, 1.18)$, $G(-6.98 \cdot 10^{-4}, 1.33, 3.50 \cdot 10^{-3}, 1.18)$ and $G(0.0189, 4.73 \cdot 10^{-3}, 0.83)$.

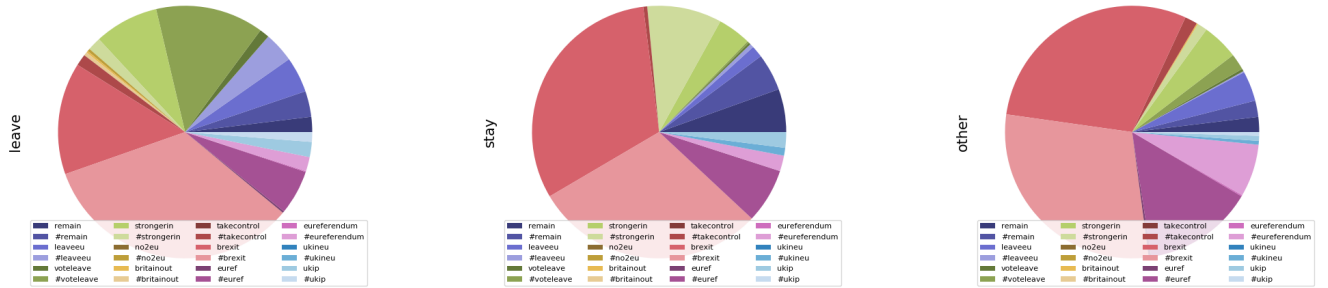


FIG. 5.— Distributions of our set of keywords in the training set Tweets with “leave”, “stay” and “other” sentiment, respectively. Tweets containing a keyword with # are only counted for the #keyword piece of this pie chart and not again for the keyword without #.

REFERENCES

- [1] C.J. Hutto, Eric Gilbert (2014) VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text <https://dev.twitter.com/rest/public/search>
- [2] VADER Sentiment Analysis, <https://github.com/cjhutto/vaderSentiment>
- [3] SSIX Brexit Gold Standard, <https://bitbucket.org/ssix-project/brexit-gold-standard>
- [4] Tweepy, <http://www.tweepy.org/>
- [5] Twitter REST APIs, <https://dev.twitter.com/rest/public/search>
- [6] Twitter Streaming APIs, <https://dev.twitter.com/streaming/overview>
- [7] Clare Llewellyn, Laura Cram (2016) Brexit? Analyzing Opinion on the UK-EU Referendum within Twitter

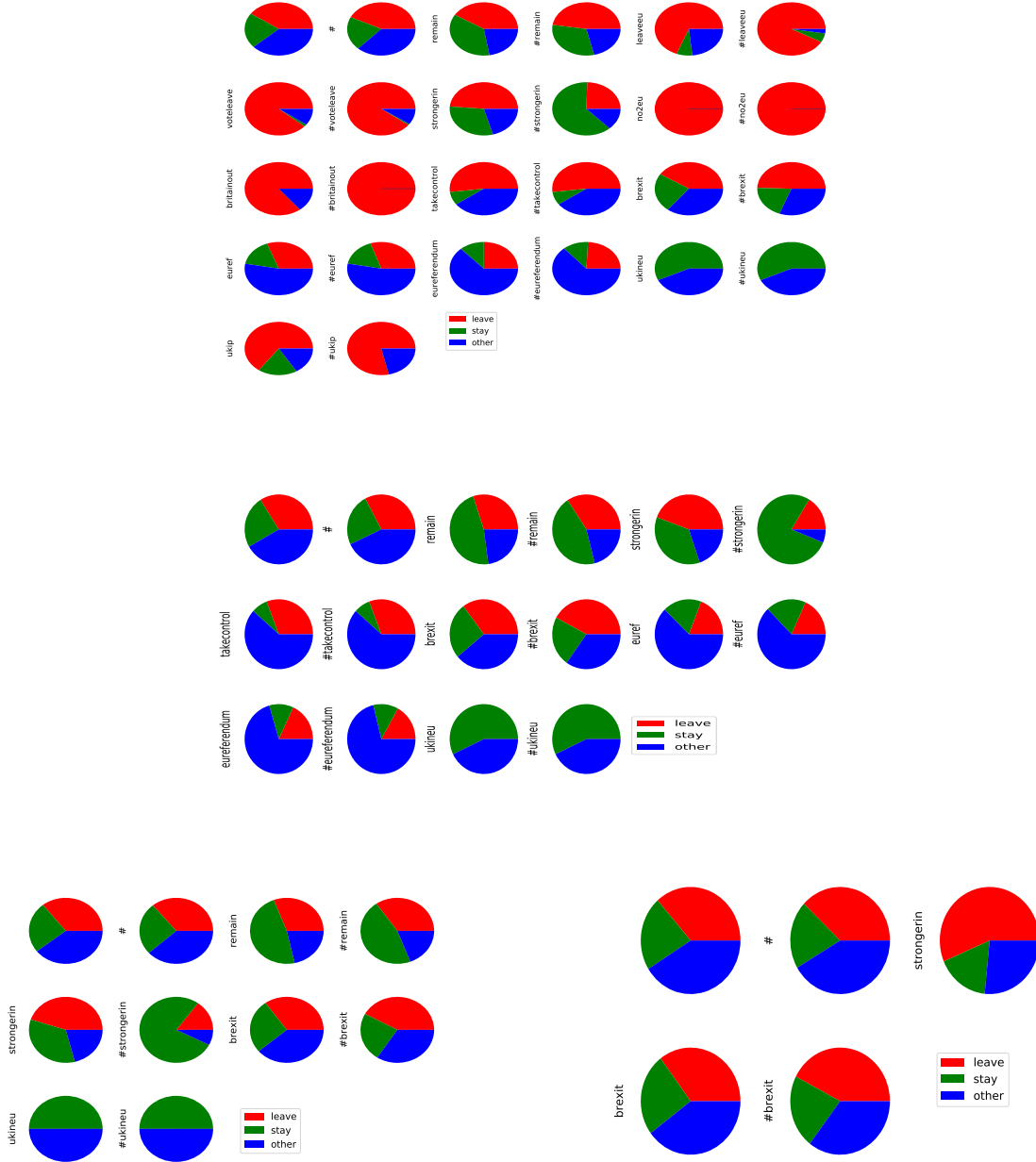


FIG. 6.— The first plot shows the sentiment distributions per keyword of the entire training set. Obviously, the keys `leaveeu`, `#leaveeu`, `voteleave`, `#voteleave`, `no2eu`, `#no2eu`, `britainout`, `#britainout`, `ukip` and `#ukip` can be associated with the sentiment “leave”. After removing all Tweets containing these classifiers we are left with the distributions in the second plot. Note, that the percentages have changed significantly, as Tweets can contain more than a single keyword. E.g.: If a Tweet contains `britainout`, `takecontrol` and `strongerin` it is already classified as “leave”, and not considered anymore. In the second run we remove Tweets with the keywords: `eurf`, `eureferendum` and `takecontrol`. Using a keyword as categoriser includes using the `#`-version as the same categoriser. This goes on until the last plot, where no clear bias is observed. For the remaining Tweets we will use our sentiment analysis.

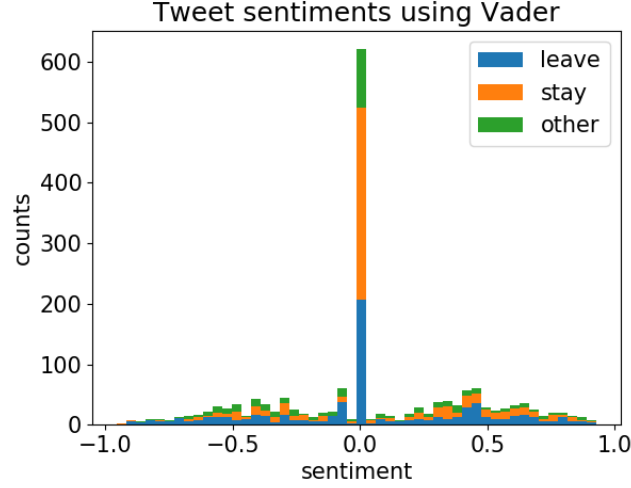


FIG. 7.— Histogram of the sentiment values assigned by VADER. There is a high peak for zero sentiment. The sentiments are almost equally distributed over the whole range.

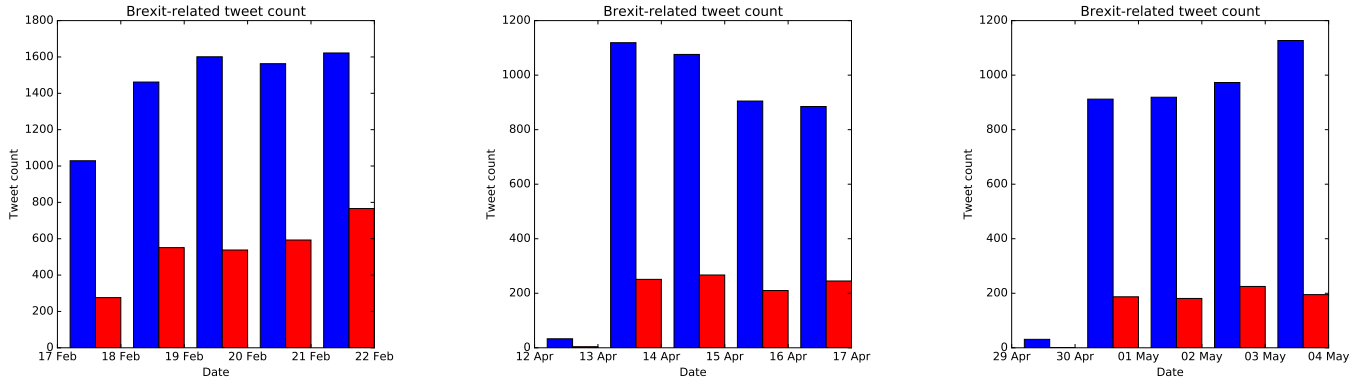
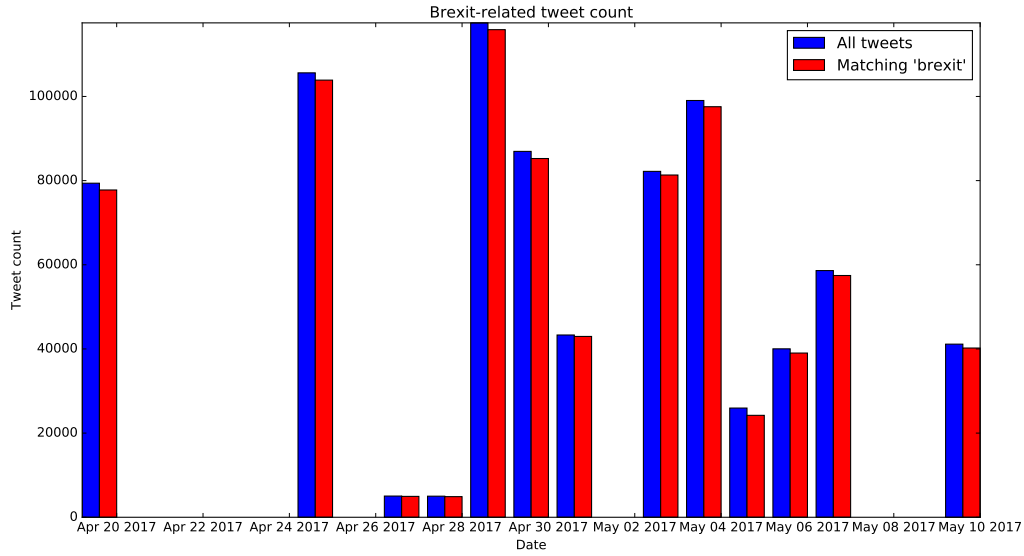


FIG. 8.— The total amount of Tweets per day. Upper plot is created using our data from April to May 2017 and the ones below using the data provided by the COSS group at ETH, which was using the Streaming APIs for data mining. We see, that the Tweet count of the latter is much lower than the first, by about a factor 100. As we know that the Streaming APIs give access to ca. 1% of the Tweets, we conclude that the REST APIs grant temporal access to the whole public data stream.

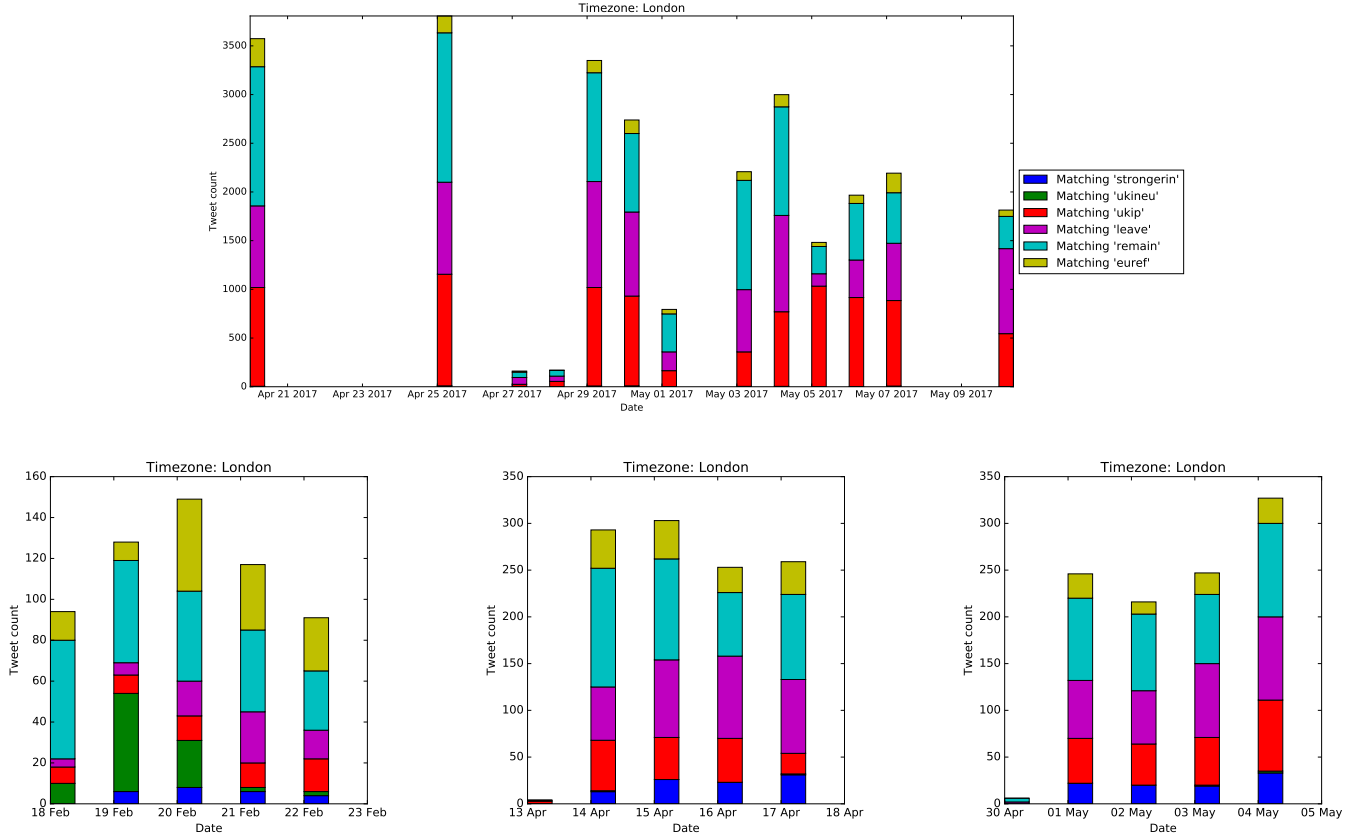


FIG. 9.— Keyword composition of the Tweets for the `user.time.zone` London, for April to May 2017 (upper plot) and certain dates in February, April and May 2016 (lower plots).

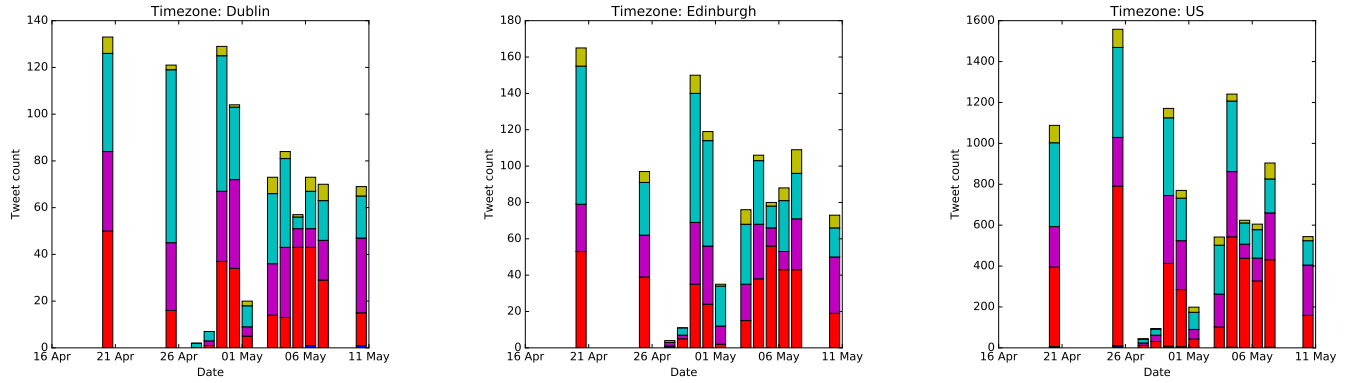


FIG. 10.— Keyword composition of the Tweets for the time zones Dublin, Edinburgh and US for late April and beginning of May 2017.

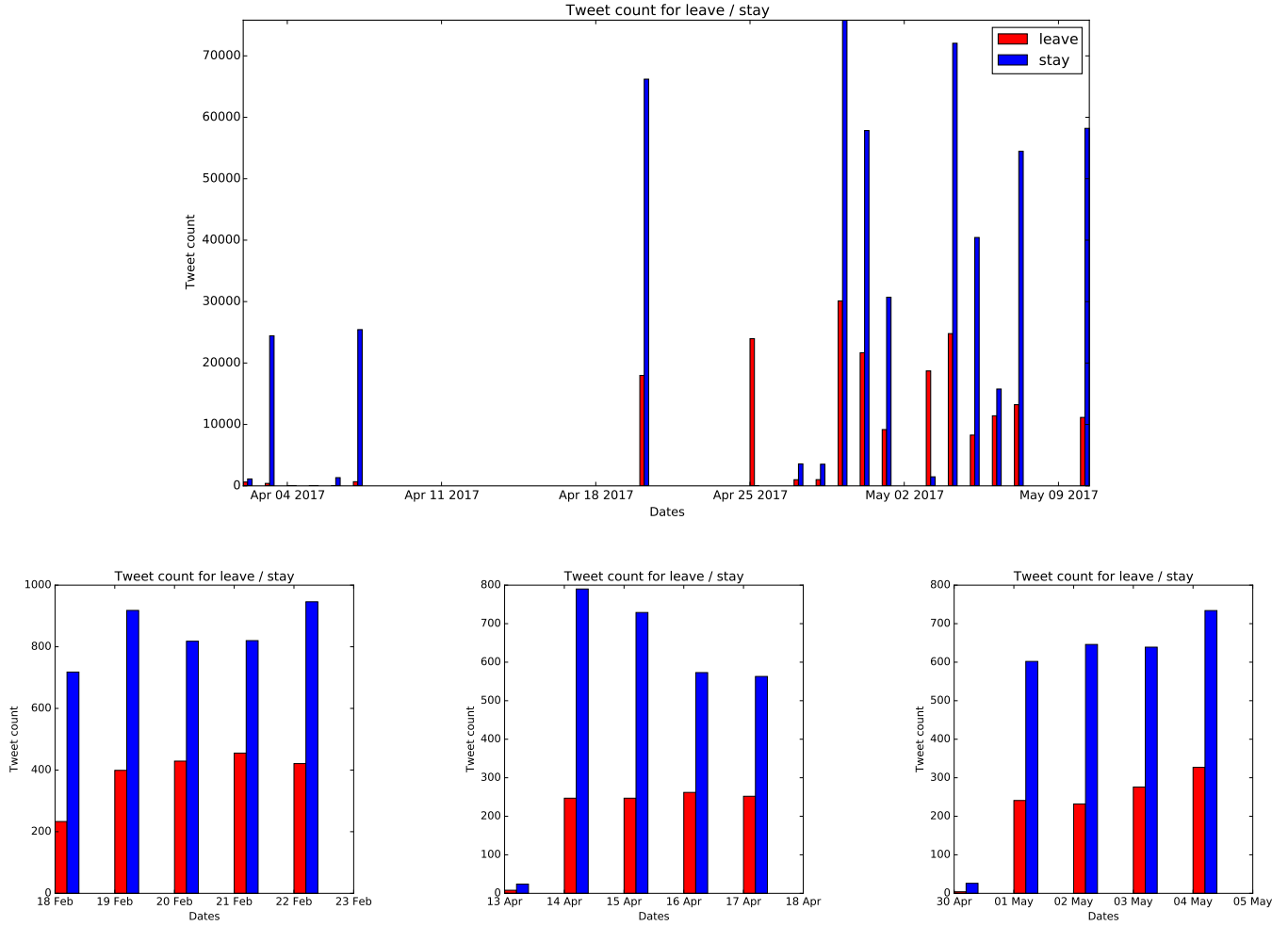


FIG. 11.— Day-for-day comparison of the number of Tweets pro-Brexit and contra, as categorised by our own sentiment analyser. On the top we see April and May 2017 and on the bottom selected dates in 2016.

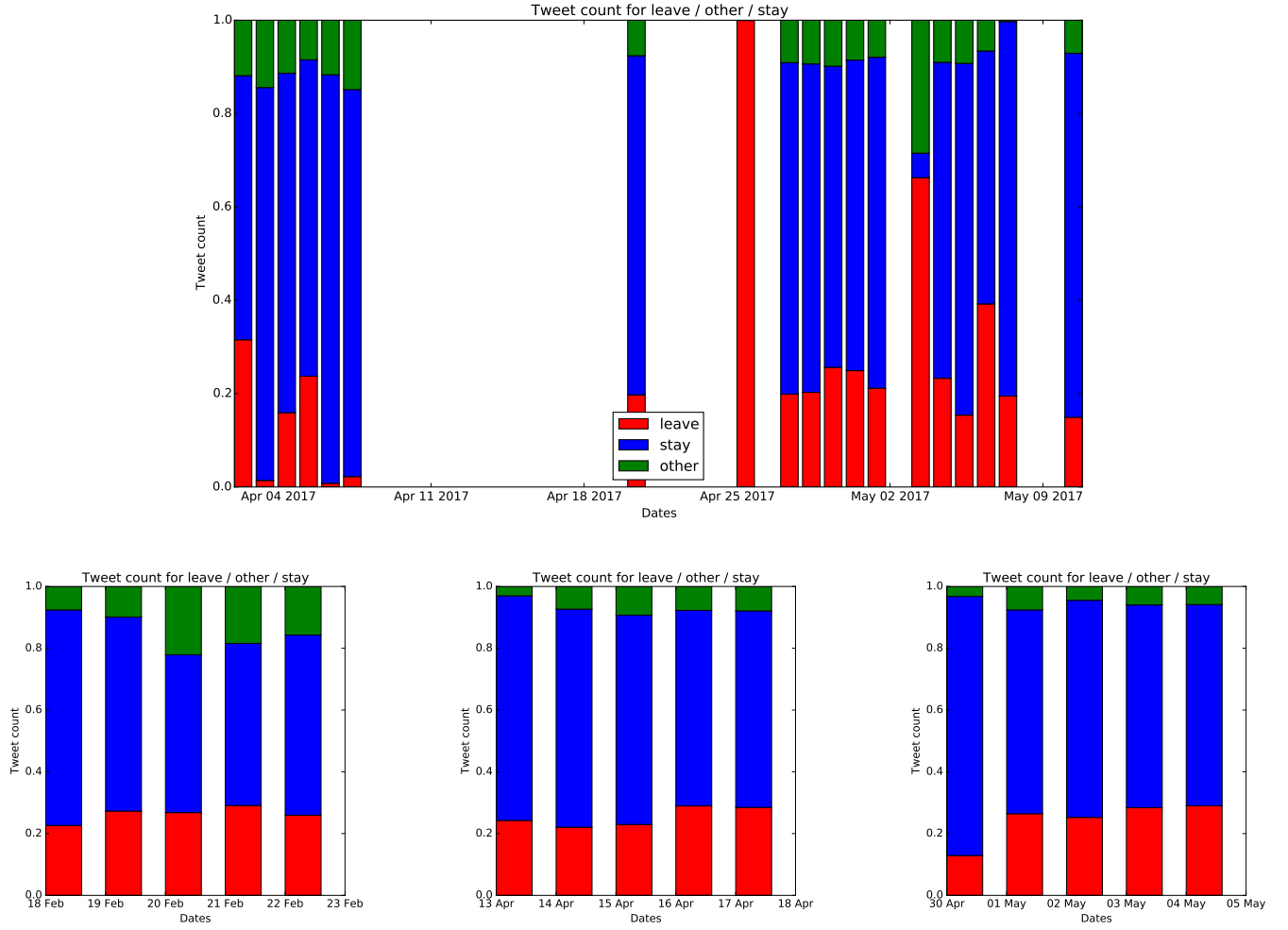


FIG. 12.— Relative day-for-day comparison of Tweets in favour of Brexit, against and without a strong sentiment, as categorised by our own sentiment analyser. On the top we see April and May 2017 and on the bottom selected dates in 2016.