Cyberpsychology, Behavior, and Social Networking

Cyberpsychology, Behavior, and Social Networking: http://mc.manuscriptcentral.com/cyberpsych

#coronavirus: Monitoring the Belgian Twitter Discourse on the Sars-Cov-2 Pandemic

Journal:	*Cyberpsychology, Behavior, and Social Networking*
Manuscript ID	CYBER-2020-0341.R1
Manuscript Type:	Original Article
Keyword:	Twitter, Quantitative Research
Manuscript Keywords (Search Terms):	Sars-coV-2, pandemic, covid-19, Twitter, Belgium

#coronavirus:

Monitoring the Belgian Twitter Discourse on the SARS-CoV-2 Pandemic

Running Title: Twitter Discourse on SARS-CoV-2

Running head: TWITTER DISCOURSE ON SARS-COV-2

Abstract

In the current study, a social media analysis is conducted to examine the public discourse about the SARS-CoV-2 pandemic on Twitter. In particular, this study aims to examine: (1) How the number of tweets varies as a function of the timeline of the pandemic and associated measures, (2) How the content of these tweets, including displayed emotions, changes. Therefore, 373,908 tweets and retweets from Belgium were collected from the 25th of February 2020 to the 30th of March. Time series analysis, network bigrams, topic models, and emotional lexica were deployed for analysis. The results showed that significant events related to the virus correlated with an immediate increase in the number of tweets addressing them. Furthermore, the Belgian Twitter discourse was characterized by positively connotated words, which also refer to European solidarity. These findings do not only stress the relevance of Twitter as a medium for public discourse during lockdowns, but also seem to indicate that the Belgian public supports policy measures which respect solidarity in Europe.

Keywords: SARS-CoV-2, pandemic, covid-19, Twitter, Belgium

Running head: TWITTER DISCOURSE ON SARS-COV-2

Introduction

The recent outbreak of the SARS-CoV-2 pandemic has disrupted the public life across the world in a manner unseen since the end of the Second World War. The virus, first identified in Wuhan (China) in December 2019, was recognized by the World Health Organization as a pandemic in March 2020 causing a severe threat to the health of people around the world. Given the serious public health risk, it was of critical importance for policy makers to respond early to be able to slow down the community spread of the virus and as such lower the burden on healthcare system. Following the examples of other countries, the Belgian government decided in several stages to take far-reaching measures to curtail public life and urge people to stay at home to further decrease the spread of the virus. In the current study, a social media analysis is conducted to examine the public discourse about the SARS-CoV-2 pandemic on Twitter. In particular, this study wants to answer two research questions: (1) How does the number of tweets vary as a function of the timeline of the pandemic and associated measures, (2) How does the content of these tweets, including displayed emotions, vary over time. Gaining profound insight in how the public perceives this pandemic and how it reacts to the measures declared by the government is of utmost importance for policymakers, health workers, and stakeholders who communicate to the public during infectious disease outbreaks.

Agenda setting theory describes the ability of media to alter the salience of issues on the public agenda[1]. According to basic agenda setting, issues that get a lot of attention in the media are considered more important by the public[2]. It has been argued that nowadays, Twitter has at least as much influence on the public agenda as newspapers [3,4]. During health crises, Twitter gives users the possibility to express opinions and share breaking news[5]. Initial studies on tweets during emerging epidemics have found that Twitter can be used to identify public concerns[6] or even trace infectious activity[7,8]. Furthermore, it can be an indicator for the public risk perception of an infection[9]. Although these initial studies report on the use of Twitter

during infectious disease outbreaks like Zika[6,9], Bird Flu (H7N9)[5] and Swine Flu (H1N1)[7,8], examining the public discourse on Twitter during the SARS-CoV-2 pandemic is particularly relevant given that the scale of it was unseen and it had an enormous impact on the daily lives of billions of people around the world.

Method

Sample

For the current study, corona-related tweets were collected from February 25, 2020 to March 30. The first patient in Belgium tested positive on SARS-CoV-2 on the 4th of February; however press attention only started to rise after about 100 Belgians were quarantined in an hotel on Tenerife, from the 25th of February onwards[10,11]. This event marks the start of our data collection. Thus, the dataset in the current study provides an overview of corona-related tweets during the unfolding of the SARS-CoV-2 pandemic in Belgium.

At the start of the data collection, it appeared that the hashtags 'coronavirus', 'covid19', 'coronavid19', and 'corona' were used most frequently in Belgium. Resultantly, these hashtags were used to collect the data. However, starting on March 11 'Coronavirusoutbreak', 'COVID19BE', and 'coronabelgie' were also regarded because Twitter displayed them as trending. Tweets were collected through the streaming Twitter API access in combination with the twitteR[12] package in R.

All tweets and retweets posted in Belgium in Dutch, French, and English language were included in the data collection, although the streaming Twitter API only allowed us to capture the first 128 characters of a sample of actual posted tweets. In total, the scraping procedure captured 373.908 tweets and retweets of which 71.481 were original tweets. A subsample of 31.454 were written in English, 15.936 in French, and 24.091 in Dutch.

Analysis

Prior to analysis, tweets were cleaned and stemmed using the text mining[13] and tidytext[14] packages. In particular, stop words, punctuation and other special signs like smileys, hashtags, slashes etc. were removed. Porter's Snowball stemmer[15] was applied to reduce the processed words to their word stems (e.g., reducing 'washing' to 'wash'). In order to answer RQ1, the number of tweets and retweets were computed on an hourly basis for the given timeframe. To tackle RQ2, three analytical techniques were applied. First, network bigrams examined which pairs of words frequently appear together and how they relate to each other. Second, to analyze the emotional connotation of the words, an emotion lexicon based on mechanical turk[16] was used. Third, machine learning based topic models were constructed to show the themes which the tweets addressed frequently. Latent dirichlet allocation was used for that purpose as it is implemented in the topic models package[17] in R.

Results

Evolution of tweets over time

Figure 1 contains the number of English, Dutch, and French tweets and retweets on the given hashtags aggregated per hour as received by the streaming API. The first peak occurs on March 2. On this day, the number of infections jumped from two to eight in Belgium[18]. Afterwards the number of tweets decreased to its initial level again until another spike occurred on Monday March 9 when stock markets crashed worldwide leading to a loss of the BEL20 of about 8%. The first death resulting from SARS-CoV-2 in Belgium was announced on the 11th of March by the Belgian Minister of Health[19]. The Government announced its decision to close schools on March 12[20], which correlated with an increase in the number of tweets. It was communicated that the schools should stay closed at least until the end of the Easter holidays on April 20. On the 14th and 15th of March the number of tweets dropped again. This is because the number of tweets is lower on the weekends. This trend of lower tweets on weekends

continues throughout the time series. On the 17th of March, stricter social distancing measures were introduced[21], meaning that leaving the place of residence was restricted to essential activities like grocery shopping, medical treatments, and exercise. Furthermore, it was generally restricted to meet in groups larger than two, except for direct relatives living in the same household. The spike on March 17 corresponds to the hour when the Belgian prime minister publicly announced the implementation of that measure. The highest number of tweets could be registered on March 19, when Michel Barnier, a well-known French politician, announced via Twitter that he had tested positive for the virus. Another rise in the number of tweets occurred one week later when the US congress approved a two trillion dollar economical stimulus[22]. On March 30, a high number of tweets addressed that the Hungarian government passed an emergency bill, resulting in a variety of executive rights for the president.

It can also be stated that the number of tweets increased rapidly after the mentioned happenings took place. Most of the tweets were sent within two hours after the occasion. The exception was the US approval of the 2 trillion dollar stimulus which became public between 5 and 6 AM Belgian time which was associated with a later spike of tweets between 10 and 11 AM.

[FIGURE 1 HERE]

Content of the tweets

For further semantic analysis, we removed the retweets from the dataset and focused on tweets in English language. We did so, because this was the most frequent language used within the captured tweets. To answer RQ2 and gain insight in the content of the tweets, a network of bigrams was computed. The network shows, which terms frequently occur together. By applying that technique, we can learn how different words are combined within tweets. As shown in Figure 2, some bigrams appear to be largely unrelated to other word pairs, e.g. ‘toilet

paper' and 'South Korea'. The main network of bigrams is centered around the words 'covid19' and 'coronavirus', containing words like 'fight', 'spread', 'response', 'test', and 'lockdown'. The network indicates some positive themes within tweets. For instance, the word 'Italian' occurs frequently together with 'support' and 'friend'. Also 'stay' and 'inside' are connected to 'safe'. Although, the word 'death' occurs in the network, there were no agonizing words like 'fear', 'disaster', or similar frequently used.

[FIGURE 2 HERE]

To gain a further understanding of word relations within the tweets, topic modelling was deployed. Its aim was to reveal several latent topics and the likelihood of words to appear within these topics. To select the appropriate number of topics, we followed the guidelines of Maier et al.[23] and not only used different metrics[24,25,26] but also tenfold cross validation. The indices showed that at least six different topics should be modelled.

[FIGURE 3 HERE]

Figure 3 displays the word stems on its y-axis and the density of terms within that topic on its x-axis represented by beta. It is not surprising that the terms 'coronavirus' and 'covid19' have a high likelihood to occur in each of the six topics. However, all of these topics stress slightly different aspects of the discourse. For instance, a tweet addressing the first topic is likely to contain the term 'crisis' ($\beta_{crisis1}$ = .0093), 'country' ($\beta_{countri1}$ = .0066), 'pandemic' ($\beta_{pandem1}$ = .0059) and stresses the role of the people to help (β_{help1} = .0062) each other and fight (β_{fight1} = .0052) the pandemic. The second topic is also likely to contain the term 'people' (β_{peopl2}

= .0061), but further stresses the need (β_{need2} = .0062) for support ($\beta_{support2}$ = .0052). It also uses the word 'health' ($\beta_{health2}$ = .0044) and addresses the situation in Italy (β_{itali2} = .0036) (which suffered the most from covid19 during the time of the data collection). The fifth topic is similar to the second one. It uses terms like 'support' ($\beta_{support5}$ = .0057) and 'itali' (β_{itali5} = .0041), although it seems to use words with a positive connotation such as 'good' (β_{good5} = .0041). The sixth and last topic addresses the need (β_{need6} = .0068) for Europe wide ($\beta_{european6}$ = .0057) testing (β_{test6} = .0067) in fighting (β_{fight6} = .0059) the spread ($\beta_{spread6}$ = .0049) of the SARS-CoV-2.

The topic models show the central motives addressed by the tweets. It can be concluded that most of them address the need for European collaboration to fight the spread of the pandemic. Several aspects of containing it are discussed in the tweets like 'the need for further testing' in topic 6 and 'the importance of time' in topic 3. Furthermore, topics 2 and 5 stress the need for international support for countries who have been severely hit by the pandemic, such as Italy.

Displayed emotions over time

Finally, the emotions associated with the words within the tweets were examined. They were coded by using an emotional lexicon by Mohammad and Turney[16]. The results are presented in Figure 4. Every layer within the area plot represents the relative amount of a given emotion within original tweets on a daily basis.

Novel to this analysis is the insight which emotions were expressed the most frequently throughout the tweets. On average, the tweets contained terms associated with 'trust' the most often ($\bar{x}_{trust}$ = 28.9 % of all words, SD_{trust} = .03), followed by 'anticipation' ($\bar{x}_{anticipation}$ = 17.5 %, $SD_{anticipation}$ = .02), and 'fear' ($\bar{x}_{fear}$ = 17.0 %, SD_{fear} = .02). Terms which were associated with 'anger' ($\bar{x}_{anger}$ = 7.8 %, SD_{anger} = .01), 'surprise' ($\bar{x}_{surprise}$ = 6.4 %, $SD_{surprise}$ = .01) and 'disgust' ($\bar{x}_{disgust}$ = 4.4 %, $SD_{disgust}$ = .01) occurred the least frequently. In addition, the analysis

showed that trust had the highest volatility among the emotions. This means that when the number of tweets increased, the proportion of trustful tweets increased more strongly than tweets expressing different emotions. The proportion of trustful tweets ranged from 21.5 % on the 29th of February to 37.1 % on the 2nd of March. Meaning that the highest proportion of tweets containing trust was obtained right after the first jump in infections. Trustful tweets decreased after both the stock market crash on the 9th of March and the decision to close schools on the 12th of March. Never-the-less, trustful tweets then rose to their second highest value on the 16th of March (33.8 %). When stricter social distancing measures were introduced on the following day, the proportion of trustful tweets decreased only slightly on 17th (30.7 %) and 18th (29.5 %) of March. When the amount of tweets peaked again on the 26th a higher proportion of trust could be detected (32 %), decreasing in the following days.

[FIGURE 4 HERE]

Discussion

The SARS-CoV-2 pandemic was the cause for a curtailment of the public life in Belgium and many countries worldwide. Governments introduced strict social distancing measures and prohibited unnecessary displacements. In these situations, social media remains one of the few areas in which public discourse is still possible. Looking at tweets from the first month of the unfolding of the pandemic, this study wanted to (1) analyze how the number of pandemic related tweets changed as a function of critical incidents, and (2) examine the content of these tweets in terms of its semantic and emotional connotation. This is especially relevant as it has been shown that Twitter is an important source of information in times of crises[27,28,29]. As such, this study provides a unique insight in the topics that were discussed by the public during the start of the SARS-CoV-2 outbreak, in the way the discourse was held, and how social media users responded to the measures taken by the government.

Time series analysis showed that significant events related to the virus resulted in an immediate increase in the number of tweets addressing them. Similar patterns have been observed in past research. Szomszor et al.[30], for instance, reported substantial increases in the number of tweets in response to a natural hazard, Terpstra et al.[28] made similar observations related to the swine flu pandemic in 2009. The current study showed that the number of tweets peaked in the hours following the governments' declaration of the measurement taken to prevent the further spread of the SARS-CoV-2, including the closing of the schools and stricter social distancing measures. In addition, high numbers of tweets were, amongst others, associated with the worldwide crash of stock markets on the 9th of March and Michel Barnier's confirmed infection on the 19th of March. These findings further stress the relevance of Twitter as a medium for public discourse during lockdowns and isolations measures; and its potential to influence the salience of the topics on the public agenda. The pace with which the discourse unfolded on Twitter after an event provided some hints for Twitter's role in the agenda setting process. Individuals' appear to turn to Twitter to communicate important events during crises. This effect could have been reinforced by the Belgian lockdown, which made it harder for citizens to discuss relevant issues offline. This implies that examining these tweets provides important information for policy makers, as the tweets help them understand the public response to policy measures.

The content of the tweets was by definition centered on the SARS-CoV-2 pandemic, but stressed different aspects. First, the bigram network and the topic models showed that the Belgian Twitter discourse was characterized by positive connotated words referring to European solidarity. For instance, 'support' and 'European' were prominent terms within the topics, as well as 'friend', 'Italian' and 'support'. These findings seem to indicate that the Belgian public supports policy measures that respect solidarity with European countries suffering from more severe consequences of the pandemic.

The conclusion that the Belgian Twitter users convey trust in their ability to overcome the pandemic is further supported by the results from the analysis of emotions within tweets. Not only was the most frequently expressed emotion 'trust', but the relative proportion of trust increased, when the number of original tweets rose. Although, previous research pointed out that Twitter can potentially create panic, e.g. during the Ebola epidemic[31], the results seem to indicate that the proportion of fear within tweets was rather low. These findings provide some evidence for Twitter's attribute agenda setting capabilities. Despite rising case numbers, the discourse emphasized trustful attributes of the topics on public agenda. We conclude that positive emotions within the Belgian Twitter discourse have the potential to increase the importance of positive aspects of the pandemic related agenda. In addition, the trust of the Belgian public in the ability to cope with the pandemic remained largely intact.

Like all research, this study has a number of limitations. First, the streaming Twitter API only provided a sample of up to 1% of all tweets. Although, it seems likely that the sample is representative, Twitter does not disclose their sampling procedure and therefore it remains unknown. Second, while English, French and Belgian sample data was collected, the analysis of the tweets content only regarded tweets in English. This might limit the possibility to infer how the Belgian Twitter uses felt as a whole. Third, Belgian Twitter users are not an accurate representation of the Belgian public, and the tweets must be regarded as an indicative sample. Fourth, it is not clear, to which extent the sample is subject to automated posting of tweets and bot activity. These issues are a common concern within the use of Twitter data[32]. Despite its limitations, the current study adds to the current state of the art Twitter analysis by being the first to examine the public discourse on Twitter in the first month of the SARS-CoV-2 crises. As such, it provides important insight into how the Belgian public responded to the far-reaching measures taken by the government to curtail public life.

Disclosure Statement

The authors declare that they have no relevant or material financial interests that relate to the research described in this paper.

References

1. McCombs ME, Shaw DL. The Agenda-Setting Function of Mass Media. Public Opinion Quarterly 1972; 36:176.

2. McCombs ME, Shaw DL, Weaver DH. New Directions in Agenda-Setting Theory and Research. Mass Communication and Society 2014; 17:781–802.

3. Valenzuela S, Puente S, Flores PM. Comparing Disaster News on Twitter and Television: an Intermedia Agenda Setting Perspective. Journal of Broadcasting & Electronic Media 2017; 61:615–37.

4. Su Y, Borah P. Who is the agenda setter? Examining the intermedia agenda-setting effect between Twitter and newspapers. Journal of Information Technology & Politics 2019; 16:236–49.

5. Vos SC, Buckner MM. Social Media Messages in an Emerging Health Crisis: Tweeting Bird Flu. Journal of health communication 2016; 21:301–8.

6. Glowacki EM, Lazard AJ, Wilcox GB, Mackert M, Bernhardt JM. Identifying the public's concerns and the Centers for Disease Control and Prevention's reactions during a health crisis: An analysis of a Zika live Twitter chat. American journal of infection control 2016; 44:1709–11.

7. Jain VK, Kumar S. An Effective Approach to Track Levels of Influenza-A (H1N1) Pandemic in India Using Twitter. Procedia Computer Science 2015; 70:801–7.

8. Signorini A, Segre AM, Polgreen PM. The use of Twitter to track levels of disease activity and public concern in the U.S. during the influenza A H1N1 pandemic. PloS one 2011; 6:e19467.

9. Chan M-PS, Winneg K, Hawkins L, Farhadloo M, Jamieson KH, Albarracín D. Legacy and social media respectively influence risk perceptions and protective behaviors during emerging health threats: A multi-wave analysis of communications on Zika virus cases. Social science & medicine (1982) 2018; 212:50–9.

10. Chini M. 110 Belgians quarantined in hotel in Tenerife due to coronavirus. Brussels Times 2020, February 25.

11. Peltier E, Minder R. Hundreds Confined to Tenerife Hotel for 14 Days Over Coronavirus Fears. The New York Times 2020, February 26.

12. Gentry J. (2016) *R Based Twitter Client*.

13. Feinerer I, Hornik K, Meyer D. Text Mining Infrastructure in R. Journal of Statistical Software 2008; 25.

14. Silge J, Robinson D. tidytext: Text Mining and Analysis Using Tidy Data Principles in R. The Journal of Open Source Software 2016; 1:37.

15. Porter MF. (2001). Snowball: A language for stemming algorithms. Available at http://snowball.tartarus.org/texts/introduction.html.

16. Mohammad S, Turney P. (2010) Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon. In: Inkpen D, Strapparava C, eds. *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text.* Los Angeles, CA: Association for Computational Linguistics, pp. 26–34.

17. Grün B, Hornik K. topicmodels : An R Package for Fitting Topic Models. Journal of Statistical Software 2011; 40.

18. Federal Public Service (FPS) Health, Food Chain Safety and Environment, Belgium. 6 new cases of Covid-19 by the end of the spring holidays. Available at https://www.info-coronavirus.be/en/news/6-new-cases-of-covid-19-by-the-end-of-the-spring-holidays/.

19. Gehrke L. Belgium confirms its first coronavirus death. Politico 2020, March 11.

20. Chini M. Closing all schools is 'out of the question', says Education Minister. Brussels Times 2020, March 12.

21. Chini M, Johnson J. Coronavirus: What are the new measures in Belgium? Brussels Times 2020, March 17.

22. Cochrane E, Fandos N. Senate Approves $2 Trillion Stimulus After Bipartisan Deal. 2020, March 25.

23. Maier D, Waldherr A, Miltner P et al. Applying LDA Topic Modeling in Communication Research: Toward a Valid and Reliable Methodology. Communication Methods and Measures 2018; 12:93–118.

24. Arun R, Suresh V, Veni Madhavan CE, Narasimha Murthy MN. (2010) On Finding the Natural Number of Topics with Latent Dirichlet Allocation: Some Observations. In: Zaki MJ, Yu JX, Ravindran B, Pudi V, eds. *Advances in Knowledge Discovery and Data Mining.* Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 391–402.

25. Cao J, Xia T, Li J, Zhang Y, Tang S. A density-based method for adaptive LDA model selection. Neurocomputing 2009; 72:1775–81.

26. Griffiths TL, Steyvers M. Finding scientific topics. Proceedings of the National Academy of Sciences of the United States of America 2004; 101 Suppl 1:5228–35.

27. Acar A, Muraki Y. Twitter for crisis communication: lessons learned from Japan's tsunami disaster. International Journal of Web Based Communities 2011; 7:392.

28. Terpstra T, Stronkman R, Vries A de, Paradies GL. (2012) Towards a realtime Twitter Analysis during crises for operational crisis management. In: Rothkrantz L, Ristvej J, Franco Z, eds. *Proceedings of the 9th International ISCRAM Conference*, pp. 1–9.

29. Wang B, Zhuang J. Crisis information distribution on Twitter: a content analysis of tweets during Hurricane Sandy. Natural Hazards 2017; 89:161–81.

30. Szomszor M, Kostkova P, Louis CS. (2011) Twitter Informatics: Tracking and Understanding Public Reaction during the 2009 Swine Flu Pandemic. In: *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*. IEEE, pp. 320–3.

31. Ahmed W, Bath PA, Sbaffi L, Demartini G. (07182018) Moral Panic through the Lens of Twitter. In: *Proceedings of the 9th International Conference on Social Media and Society*. New York, NY, USA: ACM, pp. 217–21.

32. Bruns A, Stieglitz S. Twitter data: What do they represent? it - Information Technology 2014; 56.

List of figure legends:

Figure 1: Number of tweets and retweets over time

Figure 2: Bigram Network

Figure 3: Topic Models

Figure 4: Emotions within tweets over time

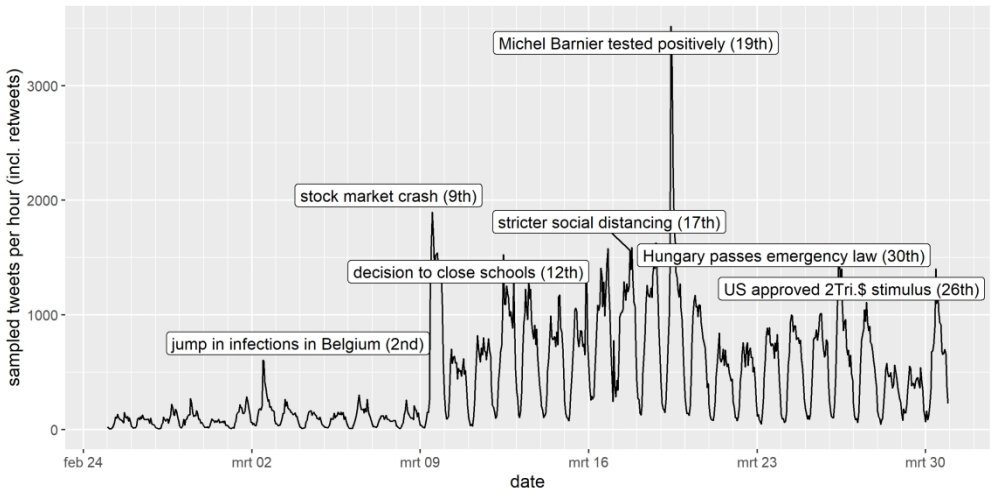


Figure 1: Number of tweets and retweets over time

238x119mm (300 x 300 DPI)

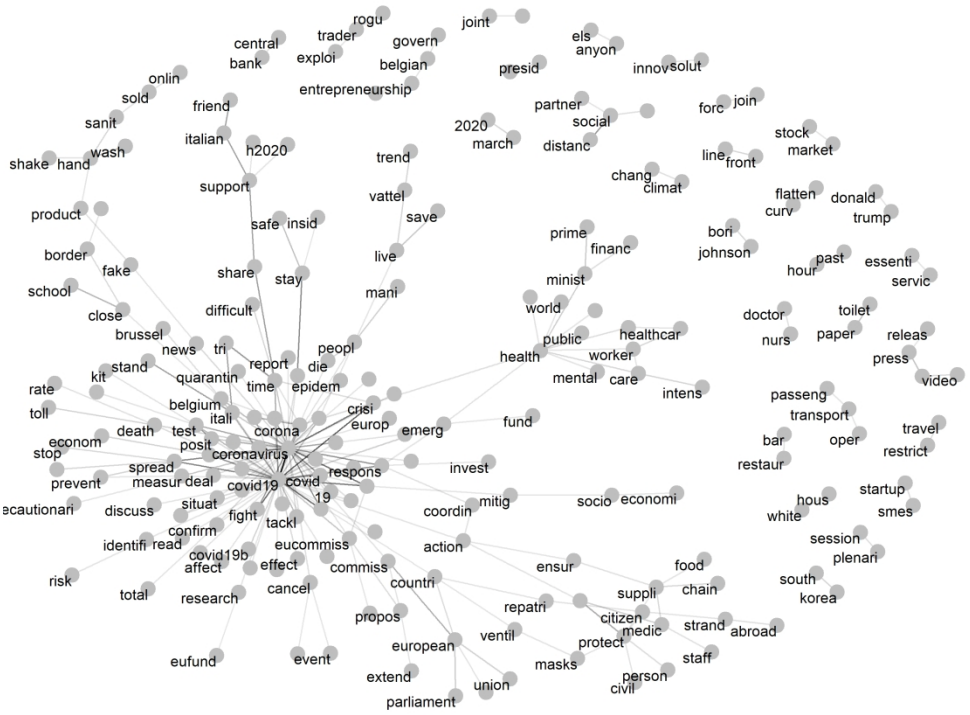


Figure 2: Bigram network

264x198mm (300 x 300 DPI)

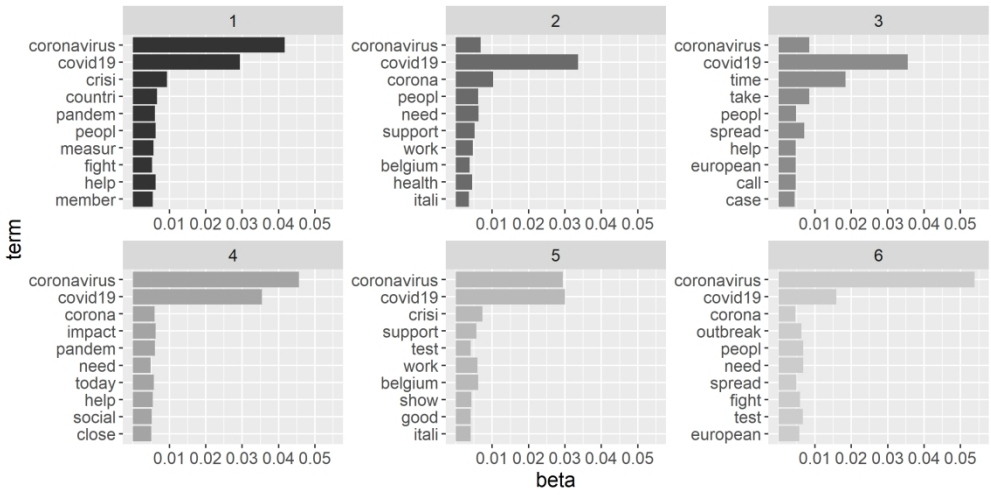


Figure 3: Topic models

238x119mm (300 x 300 DPI)

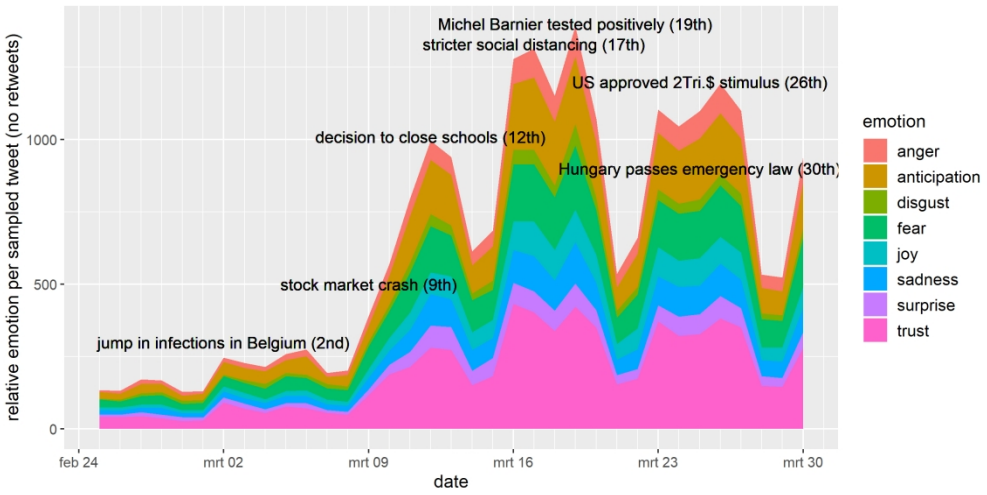


Figure 4: Emotions within tweets over time

238x119mm (300 x 300 DPI)

#coronavirus: Monitoring the Belgian Twitter Discourse on the Sars-Cov-2 Pandemic

Technical Appendix

Authors anonymous for peer review

20 July, 2020

Setup

```
library (twitteR)
library (ROAuth)
library (rtweet)
library (openxlsx)
library (dummies)
library (prodlim)
library (ggplot2)
library (dplyr)
library (tm)
library (SnowballC)
library (wordcloud)
library (RColorBrewer)
library (textdata)
library (tidytext)
library (SentimentAnalysis)
library (syuzhet)
library (reshape2)
library (tidyr)
library (igraph)
library (tidygraph)
library (ggraph)
library (topicmodels)
library (scales)
library (doParallel)
library (ldatuning)

######connect to the API#######

appname <- "corona_extraction"
key <- "XXXXXXXXXXXXXXXXX"
secret <- "XXXXXXXXXXXXXXXXXXXXXXX"
access_token <- "XXXXXXXXXXXXXXXXXXXXXXXX"
access_secret <- "XXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
credent <- c(key, secret, access_token, access_secret)

#setup_twitter_oauth(consumer_key = key, consumer_secret = secret,
```

```
access_token = access_token, access_secret = access_secret)
#commented out because this script is only used for reporting

token <- create_token(
  app = appname,
  consumer_key = key,
  consumer_secret = secret,
  access_token = access_token,
  access_secret = access_secret)
```

Hashtags to follow:

- coronavid19
- coronavirus
- covid19
- corona
- coronavirusoutbreak
- COVID19BE (from 11th March on)
- coronabelgie (from 11th March on)

We followed these hashtags from the 25th of February to the 31st of March (exclusive the 31st).

Scraping the Data

```
#####set a geocode#####

#point coordinates for belgium are
#lat       lon
#50.640281  4.666715

#A radius of 120 KM equals the size of Belgium
```

The code below for scraping is only a sample of the scraping code because it was shortened for clarity. The originally used code scraped in smaller timeframes (up to 1 day per language per hashtag per query) to not exceed the limit of the Twitter API which is 18000 tweets per query. The code below shows 15 queries. In total 120 to 150 queries were used.

```
twt_coronavid19_NL_02_25 <- twListToDF(searchTwitter("#coronavid19", since =
"2020-02-25", until = "2020-03-01",lang = "nl",n = 5000,geocode =
"50.640281,4.666715,120km"))
twt_coronavid19_NL_02_25$language <- "NL"
twt_coronavid19_FR_02_25 <- twListToDF(searchTwitter("#coronavid19", since =
"2020-02-25", until = "2020-03-01", lang = "fr", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_coronavid19_FR_02_25$language <- "FR"
twt_coronavid19_EN_02_25 <- twListToDF(searchTwitter("#coronavid19", since =
"2020-02-25", until = "2020-03-01", lang = "en", n = 5000, geocode =
"50.640281,4.666715,120km"))
```

```
twt_coronavid19_EN_02_25$language <- "EN"
twt_coronavid19_02_25 <- rbind (twt_coronavid19_NL_02_25,
twt_coronavid19_FR_02_25, twt_coronavid19_EN_02_25)
twt_coronavid19_02_25$ht_coronavid19 <- "coronavid19"

twt_coronavirus_NL_02_25 <- twListToDF(searchTwitter("#coronavirus", since =
"2020-02-25", until = "2020-03-01",lang = "nl",n = 5000,geocode =
"50.640281,4.666715,120km"))
twt_coronavirus_NL_02_25$language <- "NL"
twt_coronavirus_FR_02_25 <- twListToDF(searchTwitter("#coronavirus", since =
"2020-02-25", until = "2020-03-01", lang = "fr", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_coronavirus_FR_02_25$language <- "FR"
twt_coronavirus_EN_02_25 <- twListToDF(searchTwitter("#coronavirus", since =
"2020-02-25", until = "2020-03-01", lang = "en", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_coronavirus_EN_02_25$language <- "EN"
twt_coronavirus_02_25 <- rbind (twt_coronavirus_NL_02_25,
twt_coronavirus_FR_02_25, twt_coronavirus_EN_02_25)
twt_coronavirus_02_25$hashtag <- "coronavirus"

twt_covid19_NL_02_25 <- twListToDF(searchTwitter("#covid19", since = "2020-
02-25", until = "2020-03-01",lang = "nl",n = 5000,geocode =
"50.640281,4.666715,120km"))
twt_covid19_NL_02_25$language <- "NL"
twt_covid19_FR_02_25 <- twListToDF(searchTwitter("#covid19", since = "2020-
02-25", until = "2020-03-01", lang = "fr", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_covid19_FR_02_25$language <- "FR"
twt_covid19_EN_02_25 <- twListToDF(searchTwitter("#covid19", since = "2020-
02-25", until = "2020-03-01", lang = "en", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_covid19_EN_02_25$language <- "EN"
twt_covid19_02_25 <- rbind (twt_covid19_NL_02_25, twt_covid19_FR_02_25,
twt_covid19_EN_02_25)
twt_covid19_02_25$hashtag <- "covid19"

twt_corona_NL_02_25 <- twListToDF(searchTwitter("#corona", since = "2020-02-
25", until = "2020-03-01",lang = "nl",n = 5000,geocode =
"50.640281,4.666715,120km"))
twt_corona_NL_02_25$language <- "NL"
twt_corona_FR_02_25 <- twListToDF(searchTwitter("#corona", since = "2020-02-
25", until = "2020-03-01", lang = "fr", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_corona_FR_02_25$language <- "FR"
twt_corona_EN_02_25 <- twListToDF(searchTwitter("#corona", since = "2020-02-
25", until = "2020-03-01", lang = "en", n = 5000, geocode =
"50.640281,4.666715,120km"))
twt_corona_EN_02_25$language <- "EN"
twt_corona_02_25 <- rbind (twt_corona_NL_02_25, twt_corona_FR_02_25,
```

```
twt_corona_EN_02_25)
twt_corona_02_25$hashtag <- "corona"

twt_coronavirusoutbreak_NL_02_25 <-
twListToDF(searchTwitter("#coronavirusoutbreak", since = "2020-02-25", until
= "2020-03-01",lang = "nl",n = 5000,geocode = "50.640281,4.666715,120km"))
twt_coronavirusoutbreak_NL_02_25$language <- "NL"
twt_coronavirusoutbreak_FR_02_25 <-
twListToDF(searchTwitter("#coronavirusoutbreak", since = "2020-02-25", until
= "2020-03-01", lang = "fr", n = 5000, geocode = "50.640281,4.666715,120km"))
twt_coronavirusoutbreak_FR_02_25$language <- "FR"
twt_coronavirusoutbreak_EN_02_25 <-
twListToDF(searchTwitter("#coronavirusoutbreak", since = "2020-02-25", until
= "2020-03-01", lang = "en", n = 5000, geocode = "50.640281,4.666715,120km"))
twt_coronavirusoutbreak_EN_02_25$language <- "EN"
twt_coronavirusoutbreak_02_25 <- rbind (twt_coronavirusoutbreak_NL_02_25,
twt_coronavirusoutbreak_FR_02_25, twt_coronavirusoutbreak_EN_02_25)
twt_coronavirusoutbreak_02_25$hashtag <- "coronavirusoutbreak"

twt_scrape_02_25 <- rbind (twt_coronavid19_02_25, twt_coronavirus_02_25,
twt_covid19_02_25, twt_corona_02_25, twt_coronavirusoutbreak_02_25)

saveRDS(twt_scrape_02_25, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\scrape_02_25_to_03_01.rds")
write.xlsx(twt_scrape_02_25, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\scrape_02_25_to_03_01.xlsx")

##########remove duplicated tweets########
#only contains the first 17 columns
unique1 <- twt_scrape_02_25[,1:17]

#dupletes is a variable containing the number of the tweet it is equal to
unique1$duplete <- row.match(unique1, unique1)
#e.g. if number != duplete it is a duplicated tweet
unique1$number <- c(1:nrow(unique1))
unique1$hashtag <- twt_scrape_02_25$hashtag

#create hashtag dummy variables
unique1 <- cbind(unique1, dummy(unique1$hashtag,sep = "_"))

#unique_data  a subset of unique 1 which only contains the duplicated tweets
#sorry for the incoherent name
unique_data <- subset (unique1, unique1$duplete != unique1$number)

#unique2 only contains the original tweets which have duplicates
unique2 <- unique1[unique_data$duplete,]

#unique3 contains the first original cases with the added dummies
unique3 <- unique2
```

```
unique3$unique1_corona <- unique3$unique1_corona+unique_data$unique1_corona
unique3$unique1_coronavid19 <-
unique3$unique1_coronavid19+unique_data$unique1_coronavid19
unique3$unique1_coronavirus <-
unique3$unique1_coronavirus+unique_data$unique1_coronavirus
unique3$unique1_coronavirusoutbreak <-
unique3$unique1_coronavirusoutbreak+unique_data$unique1_coronavirusoutbreak
unique3$unique1_covid19 <-
unique3$unique1_covid19+unique_data$unique1_covid19

#create new dataframe for further operations
unique4 <- unique1

#replace the values for the dummy variables of the original variables with
the dupletes in the full dataset
unique4$unique1_corona[match(unique3$number, unique4$number)] <-
unique3$unique1_corona
unique4$unique1_coronavirus[match(unique3$number, unique4$number)] <-
unique3$unique1_coronavirus
unique4$unique1_coronavid19[match(unique3$number, unique4$number)] <-
unique3$unique1_coronavid19
unique4$unique1_coronavirusoutbreak[match(unique3$number, unique4$number)] <-
unique3$unique1_coronavirusoutbreak
unique4$unique1_covid19[match(unique3$number, unique4$number)] <-
unique3$unique1_covid19

#create a subset which does not contain the dupletes
unq_twt_02_25 <- subset(unique4, number==duplete)

#unq_twt_02_25 is now a dataframe with contains the unique tweets and the
dummy coded hashtag count
unq_twt_02_25$duplete <- NULL
unq_twt_02_25$number <- NULL
unq_twt_02_25$hashtag <- NULL

#rename the colnums to a more handy name
colnames (unq_twt_02_25)[colnames(unq_twt_02_25) == "unique1_corona"] <-
"ht_corona"
colnames (unq_twt_02_25)[colnames(unq_twt_02_25) == "unique1_coronavid19"] <-
"ht_coronavid19"
colnames (unq_twt_02_25)[colnames(unq_twt_02_25) == "unique1_coronavirus"] <-
"ht_coronavirus"
colnames (unq_twt_02_25)[colnames(unq_twt_02_25) ==
"unique1_coronavirusoutbreak"] <- "ht_coronavirusoutbreak"
colnames (unq_twt_02_25)[colnames(unq_twt_02_25) == "unique1_covid19"] <-
"ht_covid19"

saveRDS(unq_twt_02_25, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_02_25_to_03_01.rds")
```

```
write.xlsx(unq_twt_02_25, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_02_25_to_03_01.xlsx")
```

Merge Scraped Data and Count Tweets

```
#merging data from the 25th February to the 31st of March

#load data and add some NAs because there was a new hashtag regarded since
scrape 03_11
unq_twt_02_25 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_02_25_to_03_01.rds")
unq_twt_02_25$ht_COVID19BE <- NA
unq_twt_02_25$ht_Coronabelgie <- NA
unq_twt_03_01 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_01_to_03_05.rds")
unq_twt_03_01$ht_COVID19BE <- NA
unq_twt_03_01$ht_Coronabelgie <- NA
unq_twt_03_05 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_05_to_03_11.rds")
unq_twt_03_05$ht_COVID19BE <- NA
unq_twt_03_05$ht_Coronabelgie <- NA
unq_twt_03_11 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_11_to_03_13.rds")
unq_twt_03_13 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_13_to_03_19.rds")
unq_twt_03_19 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_19_to_03_23.rds")
unq_twt_03_23 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_23_to_03_26.rds")
unq_twt_03_26 <- readRDS (file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\unq_twt_03_26_to_03_31.rds")

twts <- rbind (unq_twt_02_25, unq_twt_03_01, unq_twt_03_05, unq_twt_03_11,
unq_twt_03_13, unq_twt_03_19, unq_twt_03_23, unq_twt_03_26)
twts_nort <- subset (twts, twts$isRetweet==FALSE)
twts_nort_EN <- subset (twts_nort, twts_nort$language=="EN")
twts_nort_FR <- subset (twts_nort, twts_nort$language=="FR")
twts_nort_NL <- subset (twts_nort, twts_nort$language=="NL")

#converting
twts$day_post <- NA
twts$day_post <- substr (twts$created,1,10)
twts$day_post <- as.Date (twts$day_post, format =  "%Y-%m-%d")

#count the number of tweets per hour
twts$hour_post <- NA
twts$hour_post <- substr (twts$created,1,13)
twts_per_hour <- twts %>% count(twts$hour_post)
twts_per_hour$`twts$hour_post` <- as.POSIXct (twts_per_hour$`twts$hour_post`,
format = "%Y-%m-%d %H") #tweets within the hour
```

```
#count the numbers of tweets per day
twts_per_day <- twts %>% count(twts$day_post)
twts_per_day$`twts$day_post` <- as.Date (twts_per_day$`twts$day_post`, format
=  "%Y-%m-%d")
```

Data Cleaning and Stemming

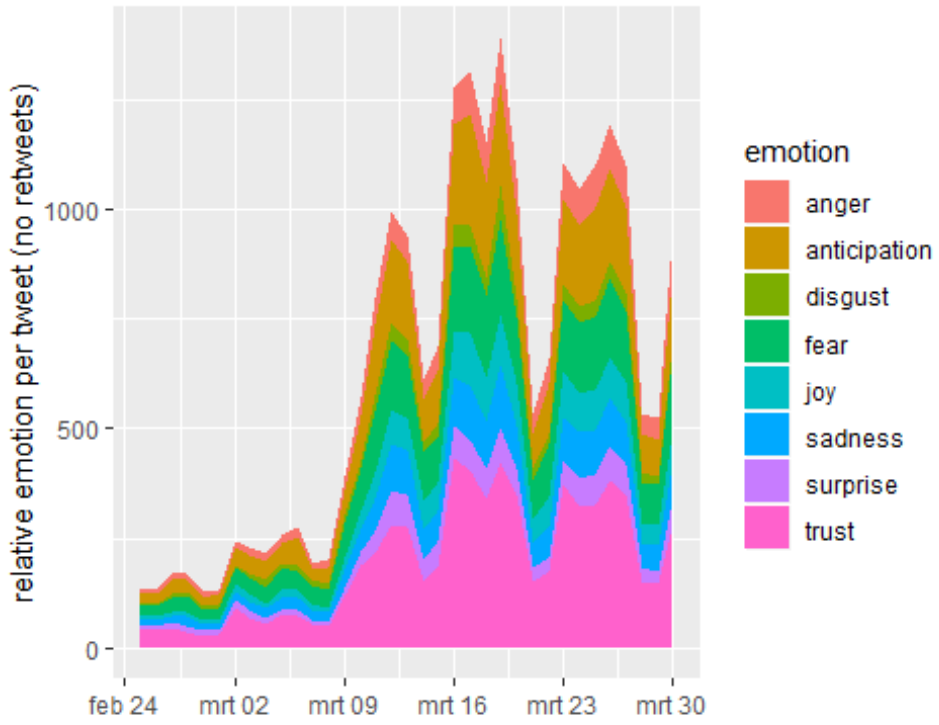
```
docs <- Corpus(VectorSource(twts_nort_EN$text))

######cleaning######

toSpace <- content_transformer(function (x , pattern ) gsub(pattern, " ", x))
docs <- tm_map(docs, toSpace, "/")
docs <- tm_map(docs, toSpace, "@")
docs <- tm_map(docs, toSpace, "\\|")
docs <- tm_map(docs, toSpace, "…")

docs[[27]]$content

## [1] "Nasa images show China pollution clear amid slowdown due to
#coronavirus. https:  t.co esQKoahPjH"

docs <- tm_map(docs, stripWhitespace)
docs <- tm_map(docs, removeWords, c("https", "t.co", "tco", "..."))
docs <- tm_map(docs, stripWhitespace)
docs <- tm_map(docs, removeWords, c("https", "t.co", "tco", "..."))
docs <- tm_map(docs, stripWhitespace)
docs <- tm_map(docs, removeWords, stopwords("english"))
docs <- tm_map(docs, stripWhitespace)
docs <- tm_map(docs, removePunctuation)
docs <- tm_map(docs, stripWhitespace)
docs <- tm_map(docs, content_transformer(tolower))
docs <- tm_map(docs, stemDocument)
docs <- tm_map(docs, stripWhitespace)

docs[[27]]$content

## [1] "nasa imag show china pollut clear amid slowdown coronavirus
esqkoahpjh"

dtm <- TermDocumentMatrix(docs)
```

Emotions over Time

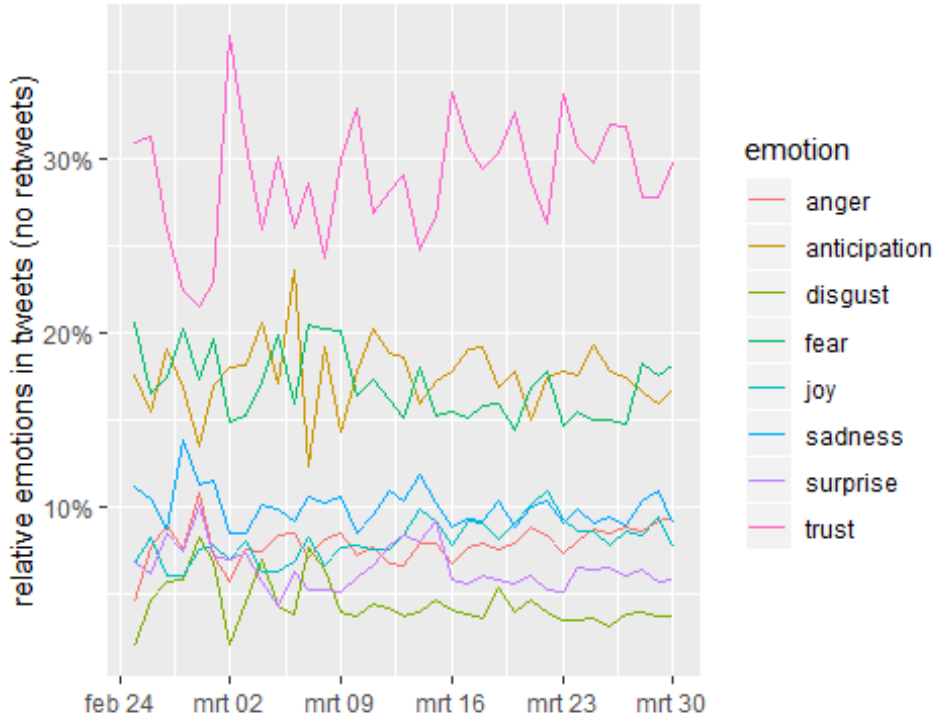
```
twts_nort_EN_sent <- twts_nort_EN

#get the emotions
sent2 <- get_nrc_sentiment(twts_nort_EN$text)

#bind it back together
```

```
twts_nort_EN_sent <- cbind (twts_nort_EN_sent, sent2)

saveRDS(twts_nort_EN_sent, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\twts_nort_EN_sent.rds")

twts_nort_EN_sent <- readRDS(file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\twts_nort_EN_sent.rds")

#####"absolute" timeseries####

attach(twts_nort_EN_sent)

#create relative count of emotions and sentiments per tweet
twts_nort_EN_sent$emotions_sum <- NA
twts_nort_EN_sent$emotions_sum <- twts_nort_EN_sent$anger +
twts_nort_EN_sent$anticipation + twts_nort_EN_sent$disgust +
twts_nort_EN_sent$fear + twts_nort_EN_sent$joy + twts_nort_EN_sent$sadness +
twts_nort_EN_sent$surprise + twts_nort_EN_sent$trust
#twts_nort_EN_sent$emotions_sum <- anger + anticipation + disgust + fear +
joy + sadness + surprise + trust

twts_nort_EN_sent$sent_sum <- negative + positive

attach(twts_nort_EN_sent)

twts_nort_EN_sent$anger_rel <- anger / emotions_sum
twts_nort_EN_sent$anticipation_rel <- anticipation / emotions_sum
twts_nort_EN_sent$disgust_rel <- disgust / emotions_sum
twts_nort_EN_sent$fear_rel <- fear / emotions_sum
twts_nort_EN_sent$joy_rel <- joy / emotions_sum
twts_nort_EN_sent$sadness_rel <- sadness / emotions_sum
twts_nort_EN_sent$surprise_rel <- surprise / emotions_sum
twts_nort_EN_sent$trust_rel <- trust / emotions_sum

twts_nort_EN_sent$positive_rel <- positive / sent_sum
twts_nort_EN_sent$negative_rel <- negative / sent_sum

#replace nan with 0 (they occured because the code above divided by 0
sometimes.)
is.nan.data.frame <- function(x)
do.call(cbind, lapply(x, is.nan))
twts_nort_EN_sent[is.nan(twts_nort_EN_sent)] <- 0

twts_nort_EN_sent$one <- 1

#maybe not necessary to convert it, because it is already in kind of a date
format.
twts_nort_EN_sent$day_post <- NA
twts_nort_EN_sent$day_post <- substr (twts_nort_EN_sent$created,1,10)
```

```
twts_nort_EN_sent$day_post <- as.Date (twts_nort_EN_sent$day_post, format =
"%Y-%m-%d")

attach(twts_nort_EN_sent)
#calculate the sum of emotions per day
anger_pd <- as.data.frame(tapply (anger_rel, day_post, sum))
anticipation_pd <- as.data.frame(tapply (anticipation_rel, day_post, sum))
disgust_pd <- as.data.frame(tapply (disgust_rel, day_post, sum))
fear_pd <- as.data.frame(tapply (fear_rel, day_post, sum))
joy_pd <- as.data.frame(tapply (joy_rel, day_post, sum))
sadness_pd <- as.data.frame(tapply (sadness_rel, day_post, sum))
surprise_pd <- as.data.frame(tapply (surprise_rel, day_post, sum))
trust_pd <- as.data.frame(tapply (trust_rel, day_post, sum))
#calculate the amount of negativity and positivity per day
negative_pd <- as.data.frame(tapply (negative_rel, day_post, sum))
positive_pd <- as.data.frame(tapply (positive_rel, day_post, sum))

emotions_pd <- cbind (anger_pd, anticipation_pd, disgust_pd, fear_pd, joy_pd,
sadness_pd, surprise_pd, trust_pd)
emotions_pd$day_post <- rownames(emotions_pd)
emotions_pd$day_post <- as.Date (emotions_pd$day_post, format =  "%Y-%m-%d")
colnames(emotions_pd) <- c("anger", "anticipation", "disgust", "fear", "joy",
"sadness", "surprise", "trust", "day_post")

emotions_plot <- pivot_longer(emotions_pd, -"day_post", names_to = "emotion")

ggplot(data = emotions_plot, aes(x=day_post, fill=emotion, y = value)) +
  geom_area() +
  scale_x_date(breaks = "week", date_labels = "%b %d") +
  xlab ("") +
  ylab ("relative emotion per tweet (no retweets)")
```




```
#####relative timeseries#####

emotions_pd_calc <- emotions_pd
#create rowsums of emotions
emotions_pd_calc$rowsum <- rowSums(emotions_pd_calc[,1:8])
#divide values by rowsums to obtain relative amount of emotion on unique
tweets within a given day
emotions_pd_rel <- sweep(emotions_pd_calc[,1:8], 1, emotions_pd_calc$rowsum,
FUN = '/')
emotions_pd_rel$day_post <- emotions_pd$day_post

emotions_rel_plot <- pivot_longer(emotions_pd_rel, -"day_post", names_to =
"emotion")

ggplot(data = emotions_rel_plot, aes(x=day_post, fill=emotion, y = value,
color = emotion)) +
  geom_line() +
  scale_x_date(breaks = "week", date_labels = "%b %d") +
  scale_y_continuous(labels = scales::percent) +
  ylab("relative emotions in tweets (no retweets)") +
  xlab ("")
```



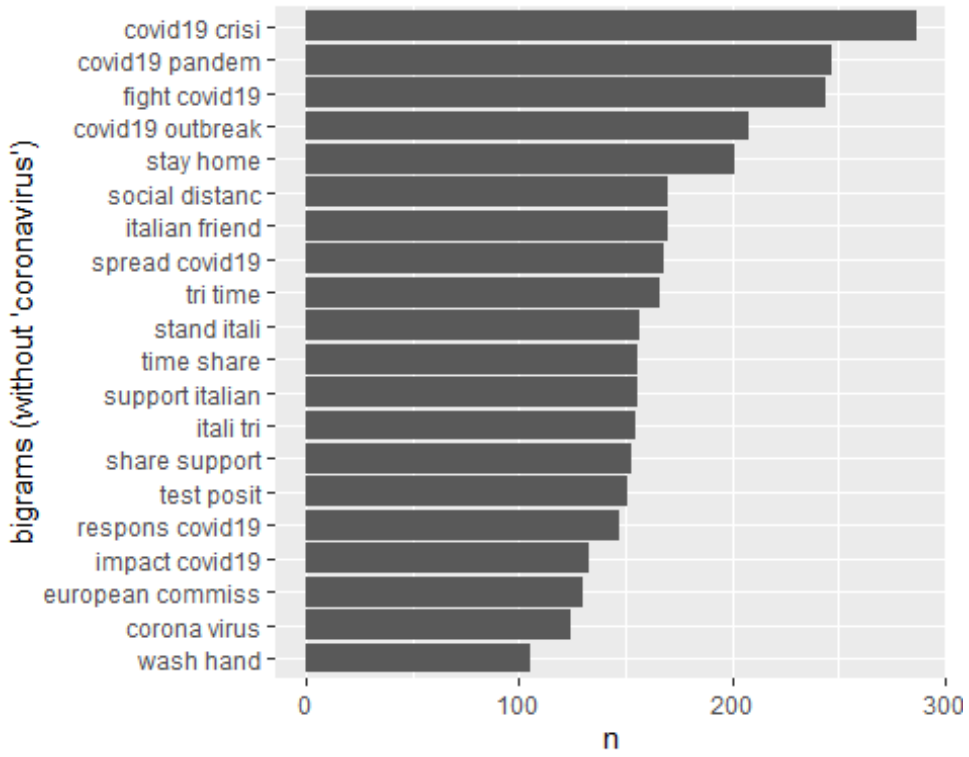
Tidy Word Analysis

```
#inspired by: https://www.tidytextmining.com/ngrams.html

#get the cleaned text from the corpus object used before
l <- length(docs)
twts_EN_nort_clean_list <- lapply(docs[1:l], as.character)
twts_EN_nort_clean <- unlist (twts_EN_nort_clean_list)

##########
twts_tidy_EN <- data.frame(txt = twts_EN_nort_clean, stringsAsFactors =
FALSE)

#frequency count of words
twts_tidy_EN %>%
  unnest_tokens(output = word, input = txt) %>%
  count(word, sort = TRUE)

## # A tibble: 69,271 x 2
##    word            n
##    <chr>       <int>
##  1 coronavirus  9274
##  2 covid19      8991
##  3 will         1887
##  4 peopl        1530
##  5 time         1441
##  6 corona       1284
```

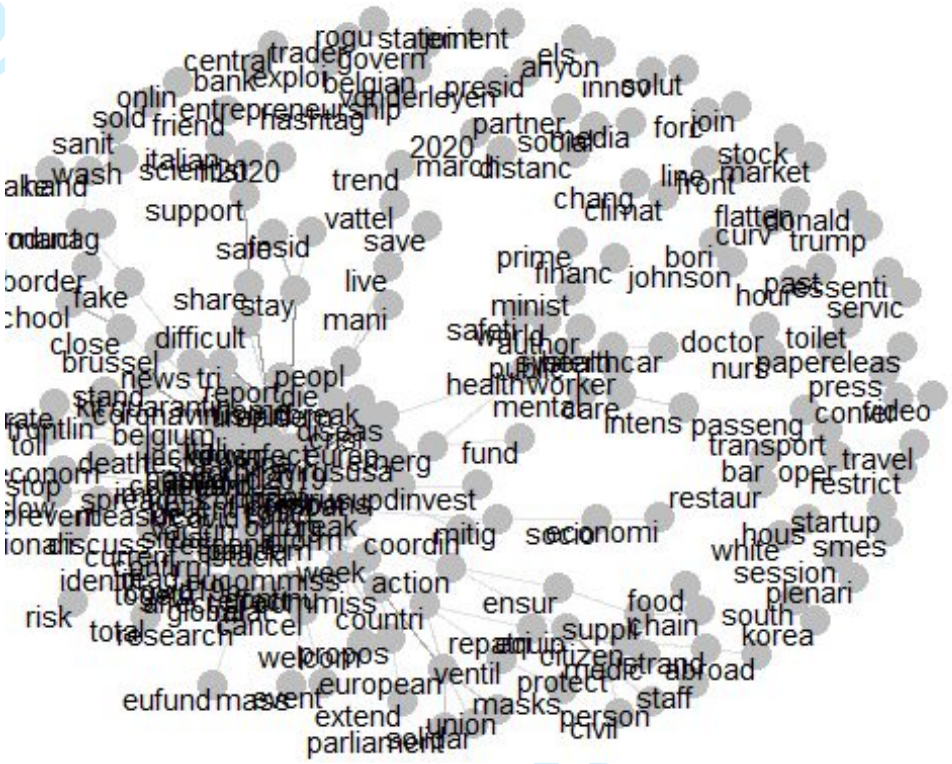
```
##  7 crisi        1092
##  8 help         1085
##  9 work         1059
## 10 need         1043
## # ... with 69,261 more rows

#remove some more stop words
cloud <- twts_tidy_EN %>%
  unnest_tokens(output = word, input = txt) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE)
cloud$word <- gsub("[^[:alnum:]]", "  ", cloud$word)
cloud <- cloud[-grep("  ", cloud$word),] #get rid of bigrams that contained
single non-alphanumeric characters (e.g. "-covid19")

#wordcloud without the words "covid19" and "coronavirus"
set.seed(2019)
wordcloud(words = cloud[3:200,]$word, freq = cloud[3:200,]$n, min.freq = 50,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```


```
bigrams <- twts_tidy_EN %>%
  unnest_tokens(bigram, txt, token = "ngrams", n=2)

bigrams <- bigrams %>%
  count(bigram, sort = TRUE)
bigrams <- as.data.frame(bigrams)
```

```
bigrams$bigram <- gsub("[^[:alnum:]]", " ", bigrams$bigram)  #get rid of all
the smilies and other non-alphanumeric characters
big_clean <- bigrams[-grep("  ", bigrams$bigram),] #get rid of bigrams that
contained single non-alphanumeric characters (e.g. "-covid19")
big_clean <- as_tibble(big_clean)
head(big_clean, 10)

## # A tibble: 10 x 2
##    bigram                  n
##    <chr>               <int>
##  1 coronavirus covid19   457
##  2 covid19 coronavirus   356
##  3 covid19 crisi         287
##  4 covid19 pandem        247
##  5 fight covid19         244
##  6 member state          241
##  7 coronavirus crisi     237
##  8 coronavirus outbreak  211
##  9 covid19 outbreak      208
## 10 stay home             201

bigrams_separated <- big_clean %>%
  separate(bigram, c("word1", "word2"), sep = " ")

#filter for stop_words
bigrams_filtered <- bigrams_separated %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word)

#create a vector which contains both words again
bigrams_filtered$words <- paste (bigrams_filtered$word1,
bigrams_filtered$word2, sep = " ", collapse = NULL)
bigrams_filtered_stripped <- bigrams_filtered[-grep("coronavirus",
bigrams_filtered$words),] #get rid of bigrams that contain the term

ggplot(bigrams_filtered_stripped[1:20,], aes(x=reorder(words, n), y=n)) +
  geom_bar(stat="identity") +
  coord_flip() +
  xlab("bigrams (without 'coronavirus')")
```




```
######visualazing bigram plots######

# filter for only relatively common combinations
bigram_graph <- bigrams_filtered %>%
  filter(n > 20) %>%
  graph_from_data_frame()

set.seed(2020)
ggraph(bigram_graph, layout = "fr") +
  geom_edge_link(aes(edge_alpha = n), show.legend = FALSE,
                 end_cap = circle(.07, 'inches')) +
  geom_node_point(color = "grey", size = 5) +
  geom_node_text(aes(label = name), vjust = 1, hjust = 1) +
  theme_void()
```



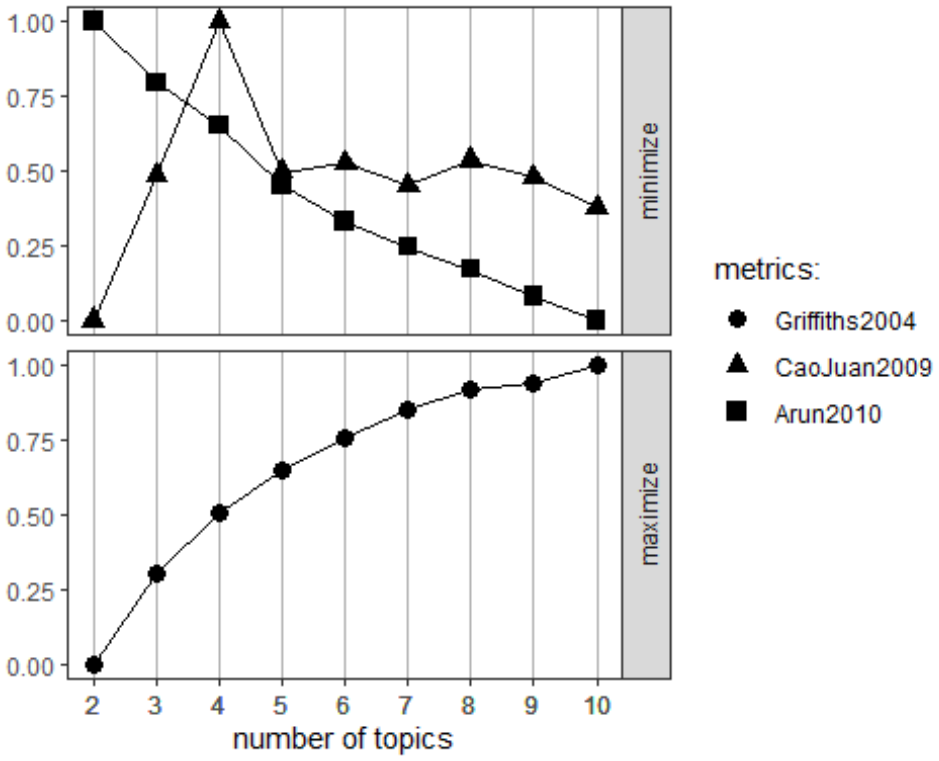
Topic Modelling

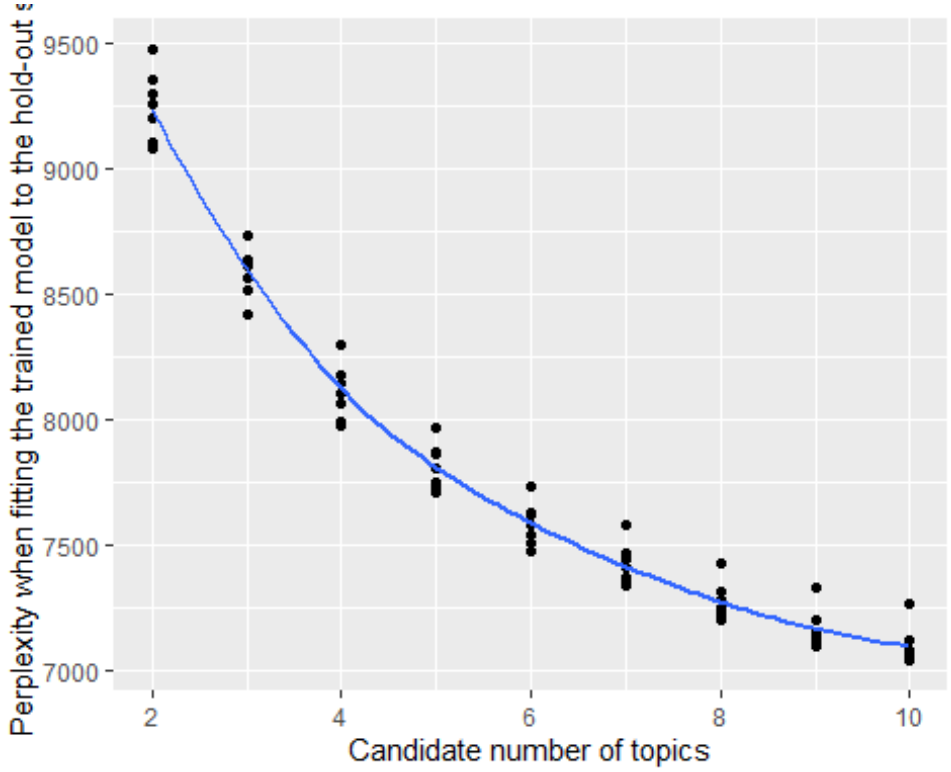
```
#inspired by: https://rpubs.com/bnevt0/AT336106

#some more cleaning for topic modelling
docs_tm <- tm_map(docs, removeWords, c("will", "just", "also", "this"))
docs_tm[[1]]$content
docs_tm <- tm_map(docs_tm, stripWhitespace)
docs_tm[[1]]$content

#create DocumentTermMatrix (different function than before)
dtm2 <- DocumentTermMatrix(docs_tm)

#Create an LDA Model, k=6 is the result of cross validation (see below)
ap_lda <- LDA(dtm2, k = 6, control = list())
ap_lda

#Extract the topics
ap_topics <- tidy(ap_lda, matrix = "beta")
ap_topics

#Show most popular per word per topic probabilities
ap_top_terms <- ap_topics %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
```

```
  arrange(topic, -beta)
ap_plot <- ap_top_terms %>%
  mutate(term = reorder(term, beta))

saveRDS(ap_plot, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\ap_plot.rds")

#####cross validation of topic models#####
#inspired by: https://rpubs.com/MNidhi/NumberoftopicsLDA

system.time({
tunes <- FindTopicsNumber(
   dtm = dtm2,
   topics = c(2:10),
   metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010"),
   method = "Gibbs",
   control = list(seed = 77),
   mc.cores = 4L,
   verbose = TRUE
)
})

saveRDS(tunes, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\tunes.rds")

#key fold cross validation

topics <- c(2:15)
burnin = 100
iter = 1000
keep = 50
folds <- 10
splitfolds <- sample(1:folds, 23, replace = TRUE)
candidate_k <- c(2:10) # candidates for how many topics

system.time({
  results <- foreach(j = 1:length(candidate_k), .combine = rbind) %dopar%{
    k <- candidate_k[j]
    results_1k <- matrix(0, nrow = folds, ncol = 2)
    colnames(results_1k) <- c("k", "perplexity")
    for(i in 1:folds){
      train_set <- dtm2[splitfolds != i , ]
      valid_set <- dtm2[splitfolds == i, ]

      fitted <- LDA(train_set, k = k, method = "Gibbs",
                    control = list(burnin = burnin, iter = iter, keep = keep)
)
      results_1k[i,] <- c(k, perplexity(fitted, newdata = valid_set))
    }
```

```
    return(results_1k)
  }
})

results_df_10fold_cv <- as.data.frame(results)
saveRDS(results_df_10fold_cv, file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\results_df_10fold_cv.rds")
```

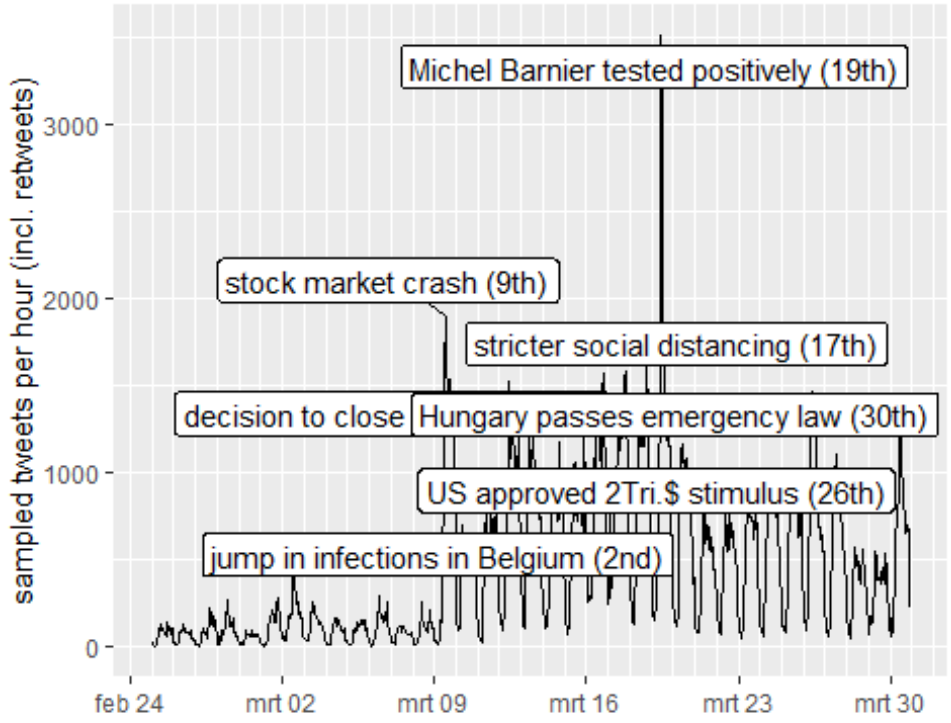
Plots to access the appropriate number of topics:


1.00
0.75
0.50
0.25
0.00
minimize
maximize
2
3
4
5
6
7
8
9
10
number of topics
metrics:
Griffiths2004
CaoJuan2009
Arun2010

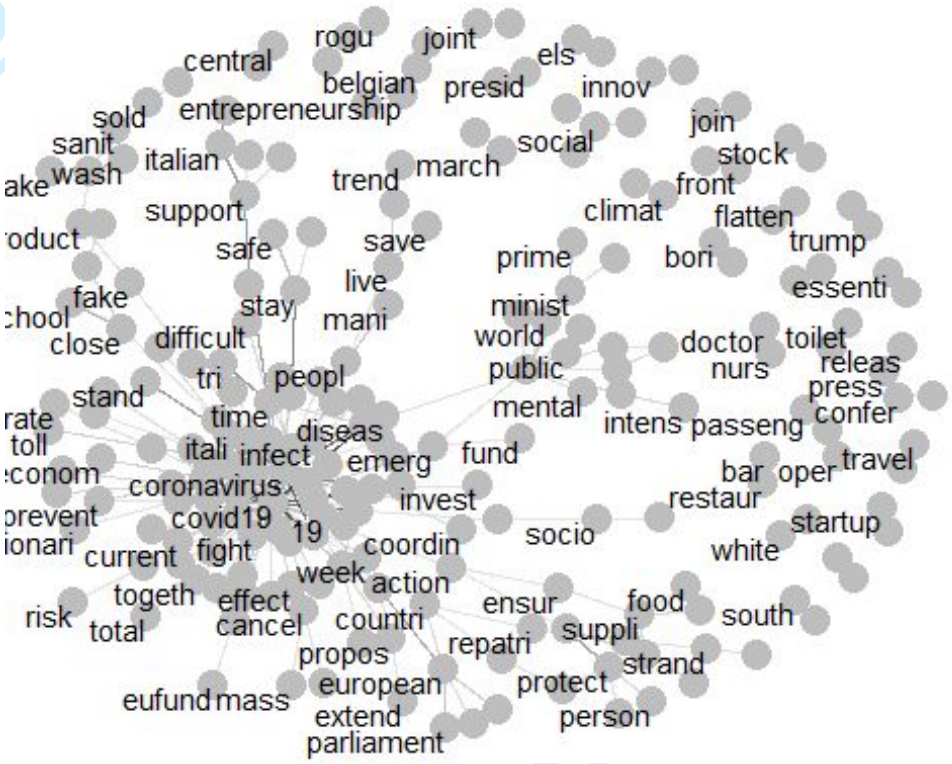

Perplexity when fitting the trained model to the hold-out s
9500
9000
8500
8000
7500
7000
2
4
6
8
10
Candidate number of topics

Reporting for the Paper

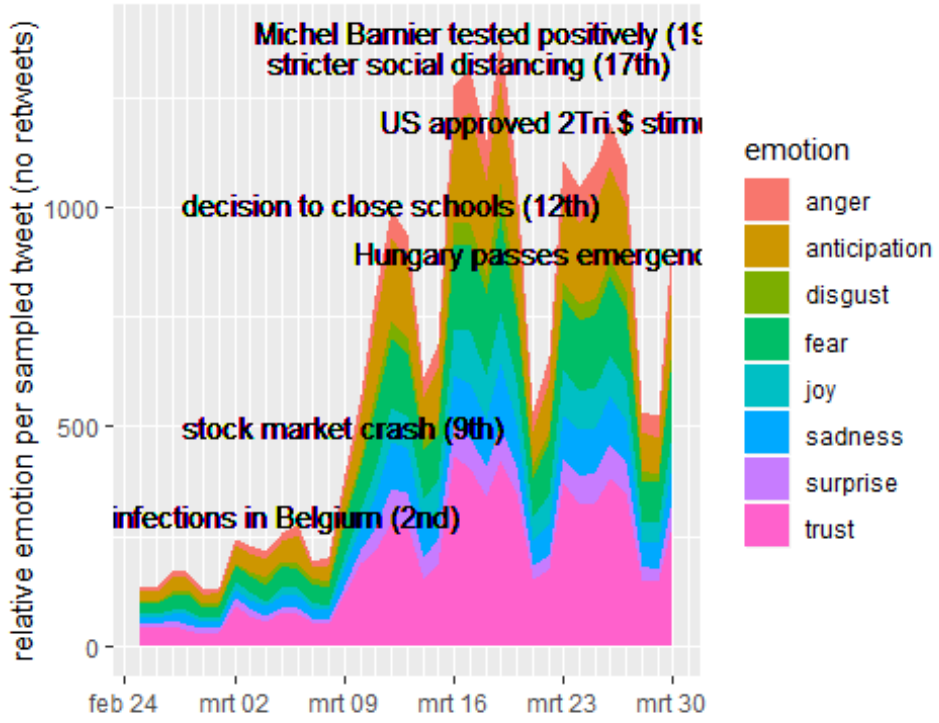
```
#####twts per hour plot

#add some labels
twts_per_hour$label <- NA
twts_per_hour$label[156] <- c("jump in infections in Belgium (2nd)") #March
2nd
twts_per_hour$label[325] <- c("stock market crash (9th)") #March 9th
twts_per_hour$label[396] <- c("decision to close schools (12th)") #March 12th
twts_per_hour$label[522] <- c("stricter social distancing (17th)") #March
17th
twts_per_hour$label[563] <- c("Michel Barnier tested positively (19th)")
#March 19th
twts_per_hour$label[729] <- c("US approved 2Tri.$ stimulus (26th)") #March
26th
twts_per_hour$label[828] <- c("Hungary passes emergency law (30th)") #March
30th

library (ggrepel)
ggplot(twts_per_hour, aes(x=`twts$hour_post`, y=n)) +
  geom_line() +
  scale_x_datetime(date_minor_breaks = "day", date_labels = "%b %d",
date_breaks = "week") +
  ylab ("sampled tweets per hour (incl. retweets)") +
  xlab ("") +
  geom_label_repel (aes (label = label))
```



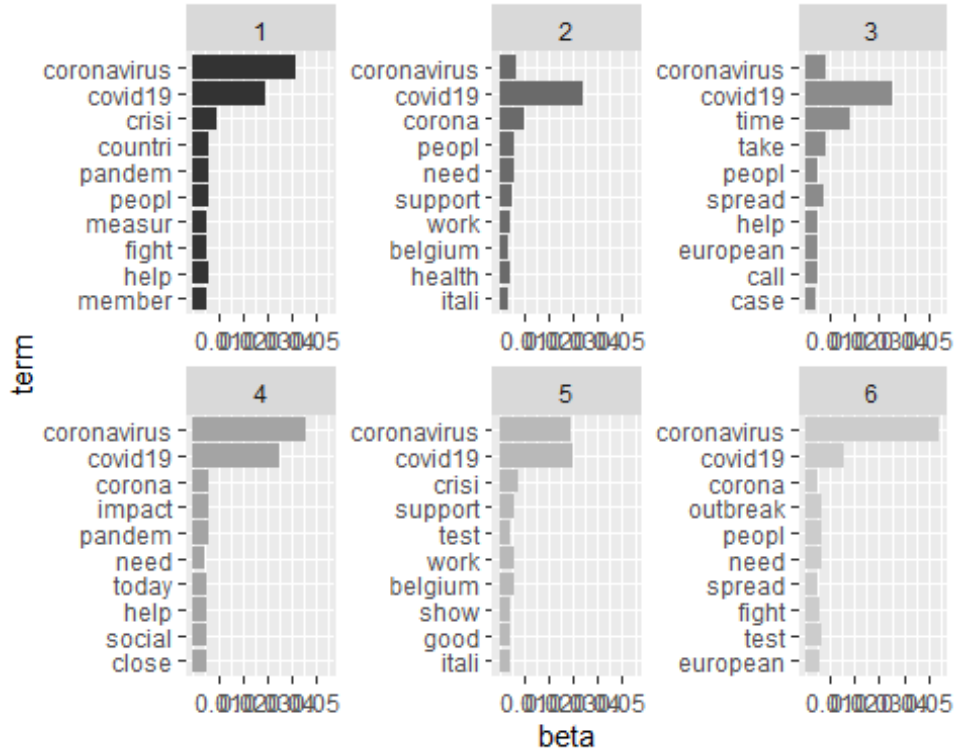

```
#####bigram network plot#####

set.seed(2020)
ggraph(bigram_graph, layout = "fr") +
  geom_edge_link(aes(edge_alpha = n), show.legend = FALSE,
                 end_cap = circle(.07, 'inches')) +
  geom_node_point(color = "grey", size = 5) +
  geom_node_text(aes(label = name), vjust = 1, hjust = 1, check_overlap =
TRUE) + #overlapping words are not plotted to increase clarity
  theme_void()
```




```
#####emotions plot######

ggplot(data = emotions_plot, aes(x=day_post, fill=emotion, y = value)) +
  geom_area() +
  scale_x_date(date_minor_breaks = "day", date_labels = "%b %d", date_breaks
= "week") +
  xlab ("") +
  ylab ("relative emotion per sampled tweet (no retweets)") +
  geom_text(x=as.Date("2020-03-02", format =  "%Y-%m-%d"), y=300,
label=c("jump in infections in Belgium (2nd)"), color="black") +
  geom_text(x=as.Date("2020-03-09", format =  "%Y-%m-%d"), y=500,
label=c("stock market crash (9th)"), color="black") +
  geom_text(x=as.Date("2020-03-12", format =  "%Y-%m-%d"), y=1010,
label=c("decision to close schools (12th)"), color="black") +
  geom_text(x=as.Date("2020-03-17", format =  "%Y-%m-%d"), y=1330,
label=c("stricter social distancing (17th)"), color="black") +
  geom_text(x=as.Date("2020-03-19", format =  "%Y-%m-%d"), y=1400,
label=c("Michel Barnier tested positively (19th)"), color="black") +
  geom_text(x=as.Date("2020-03-26", format =  "%Y-%m-%d"), y=1200,
label=c("US approved 2Tri.$ stimulus (26th)"), color="black") +
  geom_text(x=as.Date("2020-03-26", format =  "%Y-%m-%d"), y=900,
label=c("Hungary passes emergency law (30th)"), color="black") #X value was
changed to make the text fit on the plot
```




```
#####topic models#####

ap_plot <- readRDS(file =
"C:\\Users\\u0135880\\Dropbox\\Twitter_Project\\ap_plot.rds")

  ggplot(data = ap_plot, aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip() +
  scale_y_continuous(breaks=c(.01,.02,.03,.04,.05), limits = c(0,0.055)) +
  scale_fill_grey()
```




```
######reporting for paper######

#number of tweets and retweets:
nrow(twts)

## [1] 373908

#number of original tweets:
nrow(twts_nort)

## [1] 71481

#number of original tweets in English:
nrow(twts_nort_EN)

## [1] 31454

#number of original tweets in Dutch:
nrow(twts_nort_NL)

## [1] 24091

#number of original tweets in French:
nrow(twts_nort_FR)

## [1] 15936

#relative overall amount of emotions
tapply (emotions_rel_plot$value, emotions_rel_plot$emotion,  mean)
```

```
##        anger anticipation      disgust         fear          joy
sadness
##   0.07824566   0.17502977   0.04393404   0.16957330   0.08018855
0.09948661
##     surprise        trust
##   0.06430995   0.28923212

#all emotions
emotions_rel_plot

## # A tibble: 280 x 3
##    day_post   emotion       value
##    <date>     <chr>         <dbl>
##  1 2020-02-25 anger        0.0452
##  2 2020-02-25 anticipation 0.175
##  3 2020-02-25 disgust      0.0193
##  4 2020-02-25 fear         0.205
##  5 2020-02-25 joy          0.0671
##  6 2020-02-25 sadness      0.111
##  7 2020-02-25 surprise     0.0677
##  8 2020-02-25 trust        0.309
##  9 2020-02-26 anger        0.0762
## 10 2020-02-26 anticipation 0.154
## # ... with 270 more rows
```