Individual Assignment – Business Insight Report

# WhatsApp privacy policy update

Text Analytics and NLP – DAT-5317

**Professor**
Thomas Kurnicki

**Submitted by**
Elise Neumann

**Date**
February 10, 2021

# Table of Contents

# Introduction

After a rough 2020 for many, we kicked-off 2021 at COVID safe parties and with high hopes for a better year. Soon to follow was an announcement from WhatsApp which put many in a frenzy: the privacy policies were changing in February, and significantly more of users' data would be shared with Facebook. The policy was presented as 'non-elective', so many customers started looking at alternative services. In my studies at Hult, it was for the first time normal to check whether everyone would be ok to use WhatsApp as the primary communication tool within a new team, whereas this had always been assumed previously.

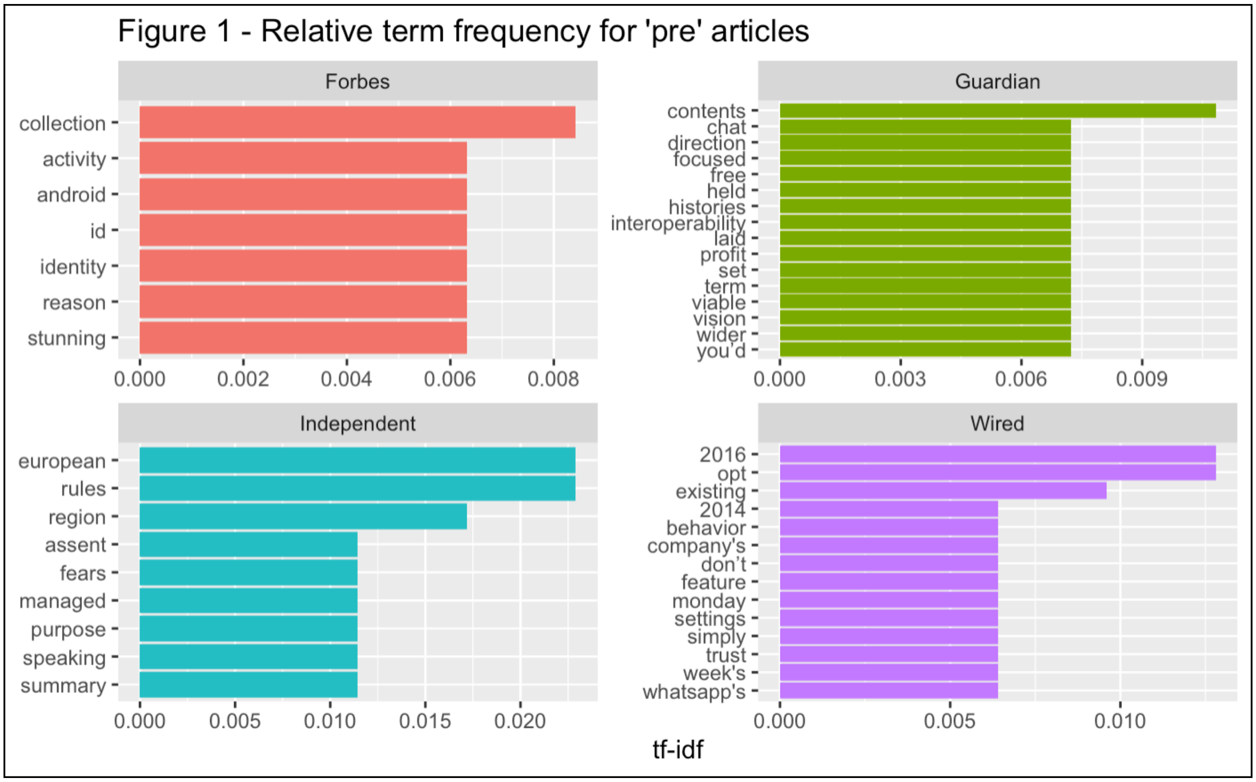
On January 12, WhatsApp released additional information about the policy update, and created a dedicated FAQ page on their site for this topic. Indeed, it would appear that the intentions, those affected, and the information which would be collected in the change were misunderstood by the general population.

In order to better understand what was being said on the topic, four articles from various sources have been collected prior to the information corrections made by Facebook, and four post January 12. Various frameworks have been applied in order to identify key topics and sentiments across articles and these two time periods.

# Articles pre correction

## Figure 1 – Term Frequency, Inverse Document Frequency

Each article takes a slightly different approach to the subject. The Independent, a British website, sees words such as ‘European’, ‘rules’ and ‘region’ highlighted as it discusses how the EU will be impacted by the update. On the other hand, Wired provides more of a timeline to the events leading up to WhatsApp’s decision, with certain dates standing out as particularly important.

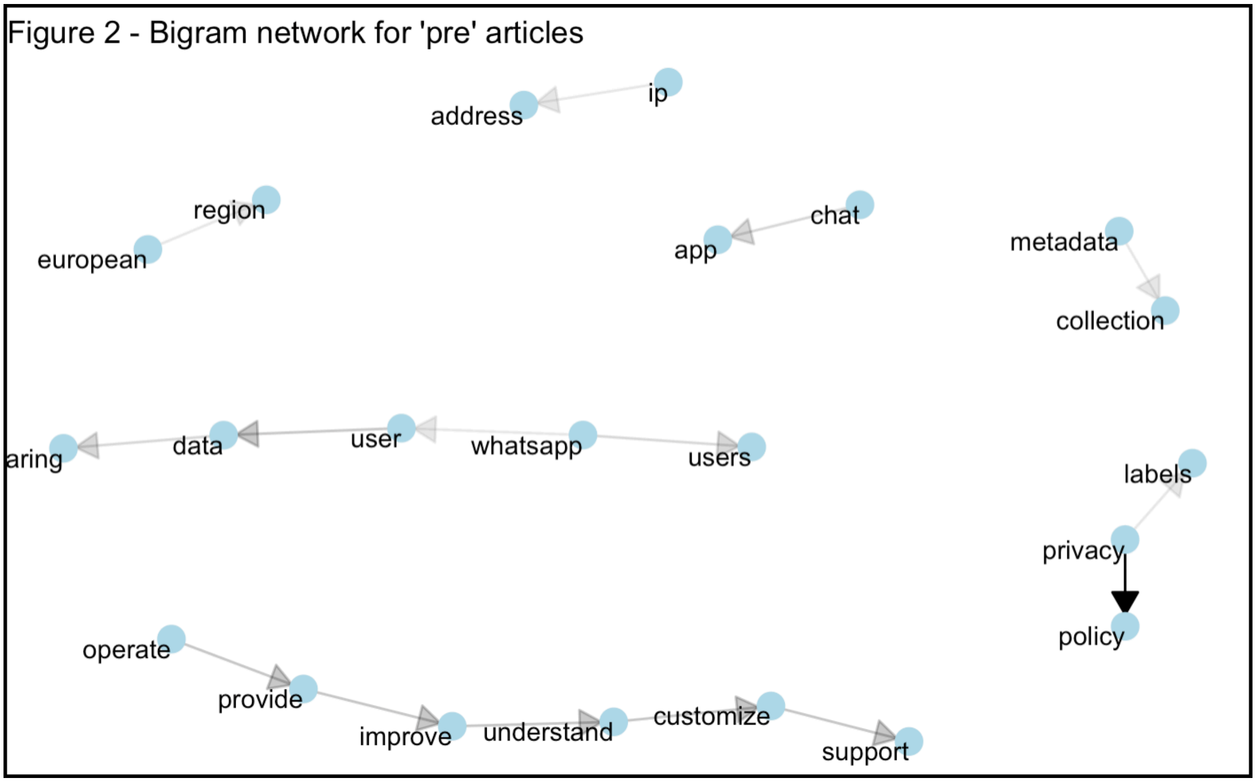


## Figure 2 – Bigram Network

As could be expected, the thickest line connects ‘privacy’ and ‘policy’, indicating that those two words are the most frequent bigram. This is in line with the topics we know to be covered in the articles.

A second chain is centered around the WhatsApp users, answering the questions who and what is at the center of the discussions in the articles.

Figure 2 - Bigram network for 'pre' articles



## Figure 3 – Sentiment Analysis

At first glance, most articles seem to have more words with positive connotations than negative. However, when looking more closely at these positive words, we realize that it may not actually be the case. For instance, ‘share’, appearing in 3/4 articles actually refers to the noun, and ‘stunning’ is used as the synonym of shocking. We can therefore conclude, as expected, that these articles convey a primarily negative sentiment, centering around fear, concern and panic.

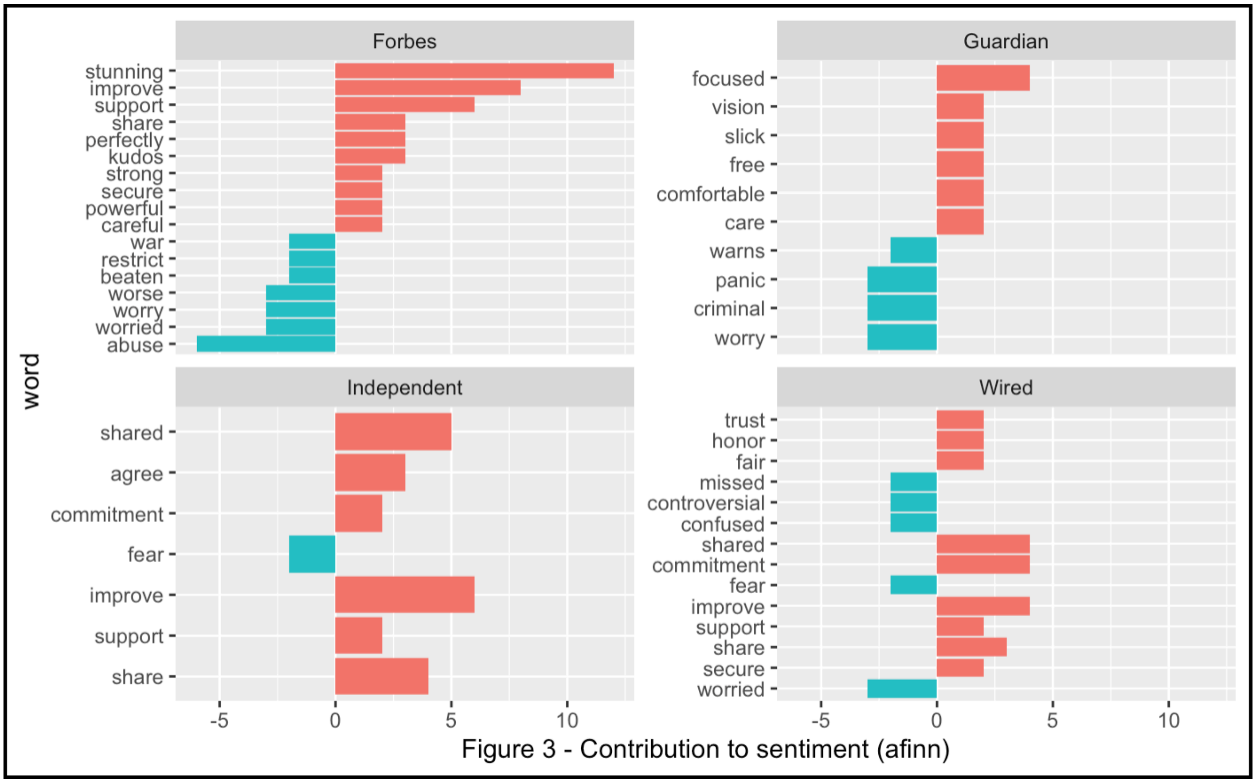
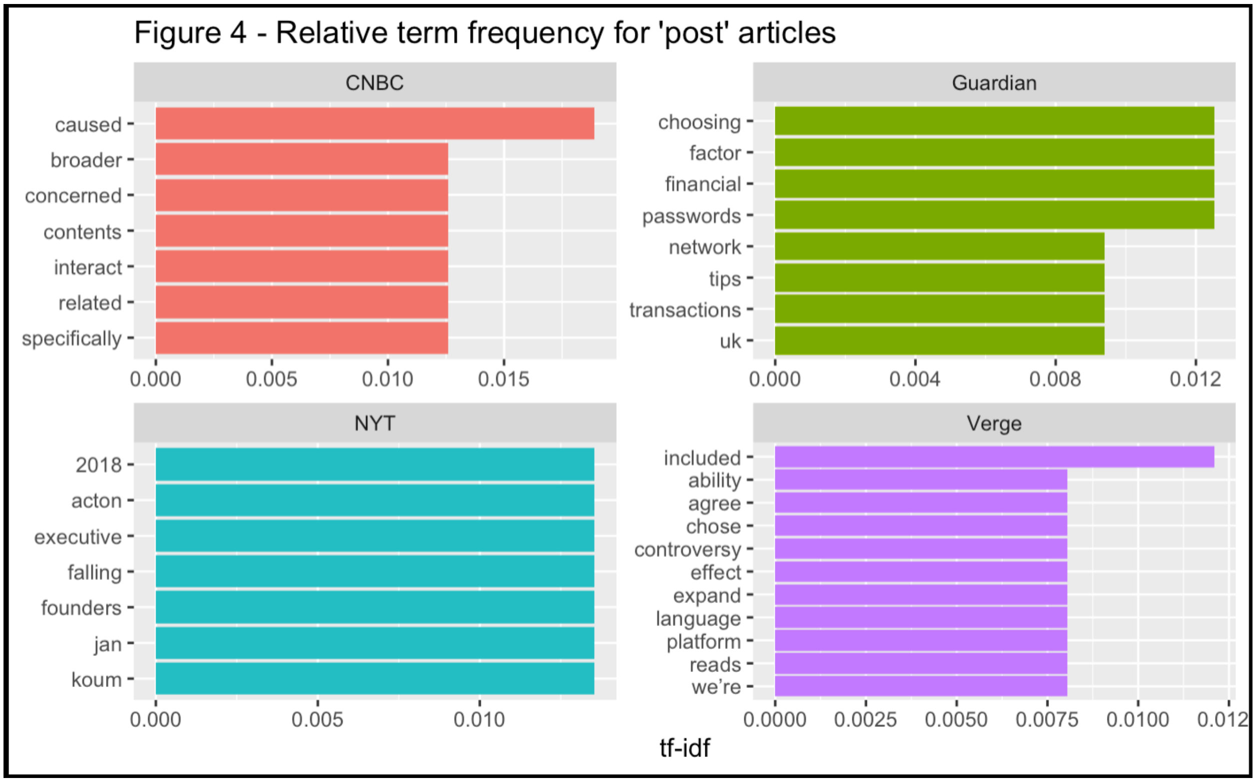


Figure 3 - Contribution to sentiment (afinn)

## Articles post correction

### Figure 4 – Term Frequency, Inverse Document Frequency

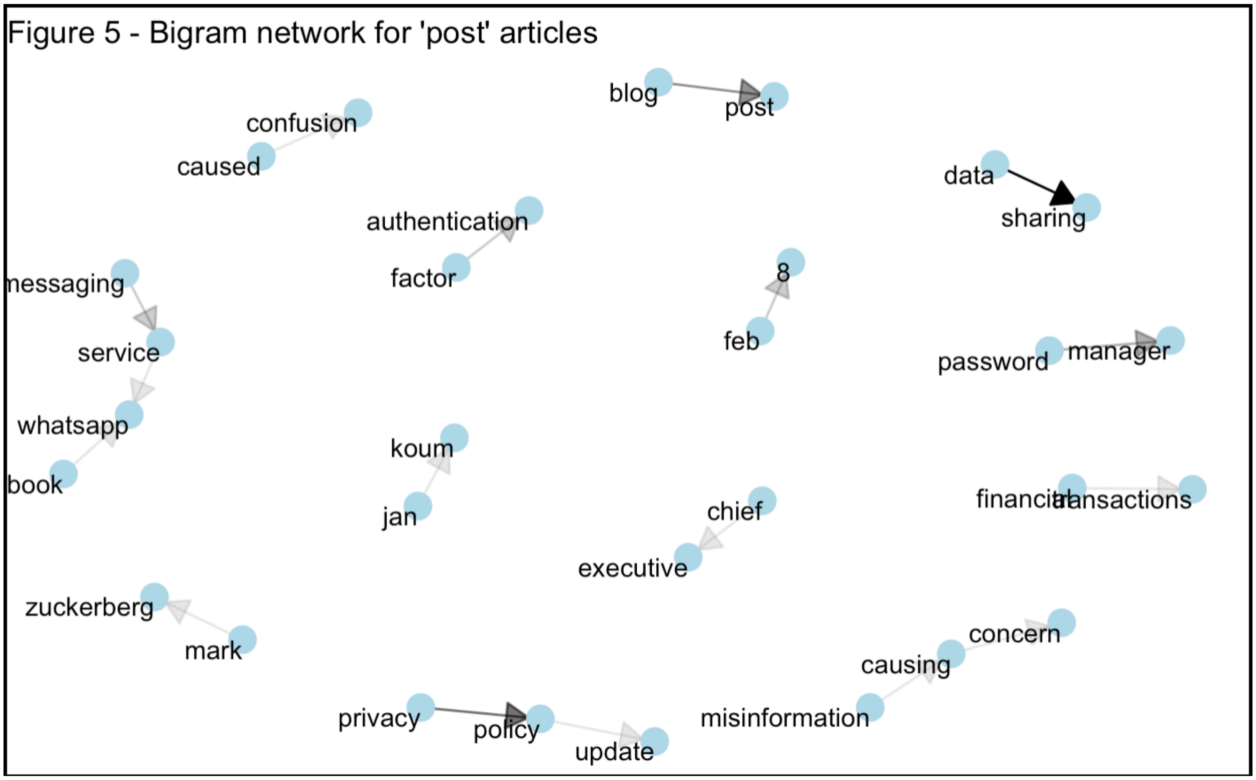
Fewer words stand out here than in the pre-announcement articles. This could be explained by the fact that the articles address the topic from a much more similar angle than they did prior to WhatsApp's clarification statement. We can wonder whether the updated information WhatsApp provided leaves less to be interpreted and questioned, so there are more similarities between articles.



## Figure 5 – Bigram Network

We want to draw your attention to a couple of the connections on the below graph. With the clarifications provided by WhatsApp, the release date, ‘Feb 8’, stands out as a recurring phrase. Unsurprisingly, reactions to the update were addressed in all the articles with the phrases ‘caused concern’ and ‘misinformation causing concern’ also standing out. Finally, ‘chief executive’ and ‘Mark Zuckerberg’ are common bigrams between the articles, as they question the decisions of key players at Facebook.

Figure 5 - Bigram network for 'post' articles



## Figure 6 – Sentiment Analysis

Once again, the articles (with the exception of NYT), appear to be primarily positive. However, considering the context of these articles, it is questionable as to whether ‘secure’, ‘share’, ‘legal’ and others truly represent positive concepts in the articles. ‘Misinformation’, significantly impacting the negative pull on 3/4 articles speaks to where some of the frustration may be coming from.

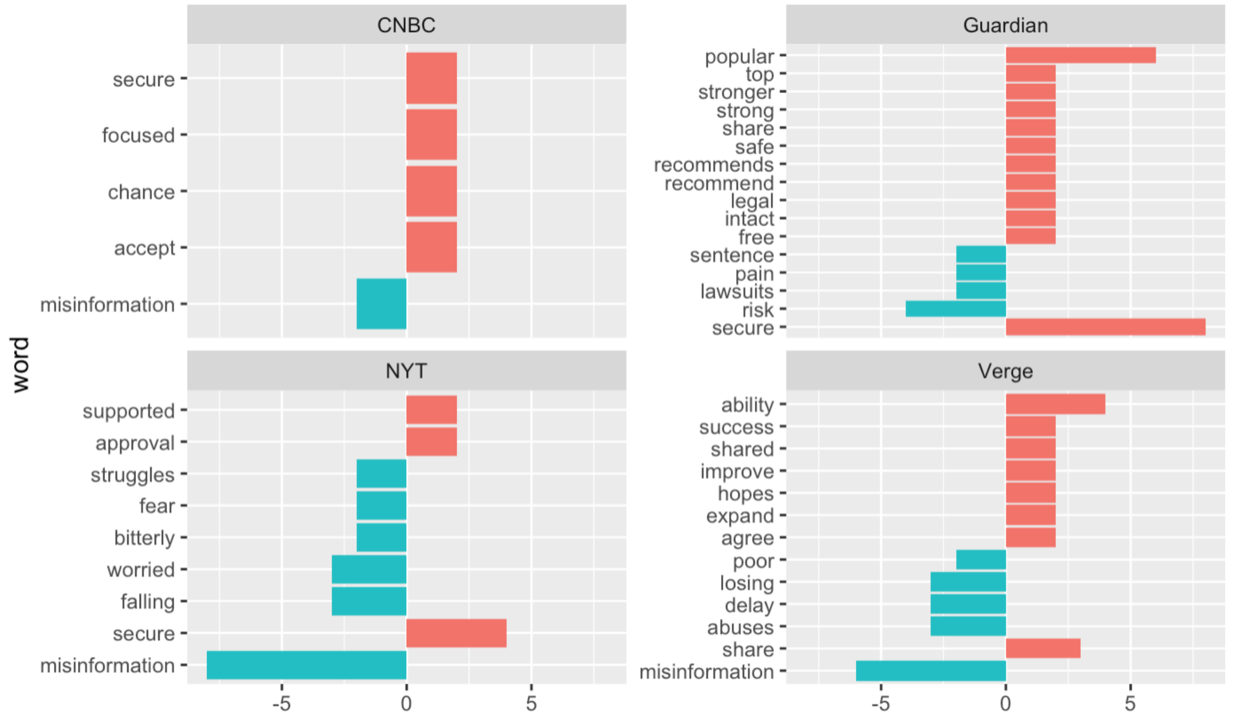
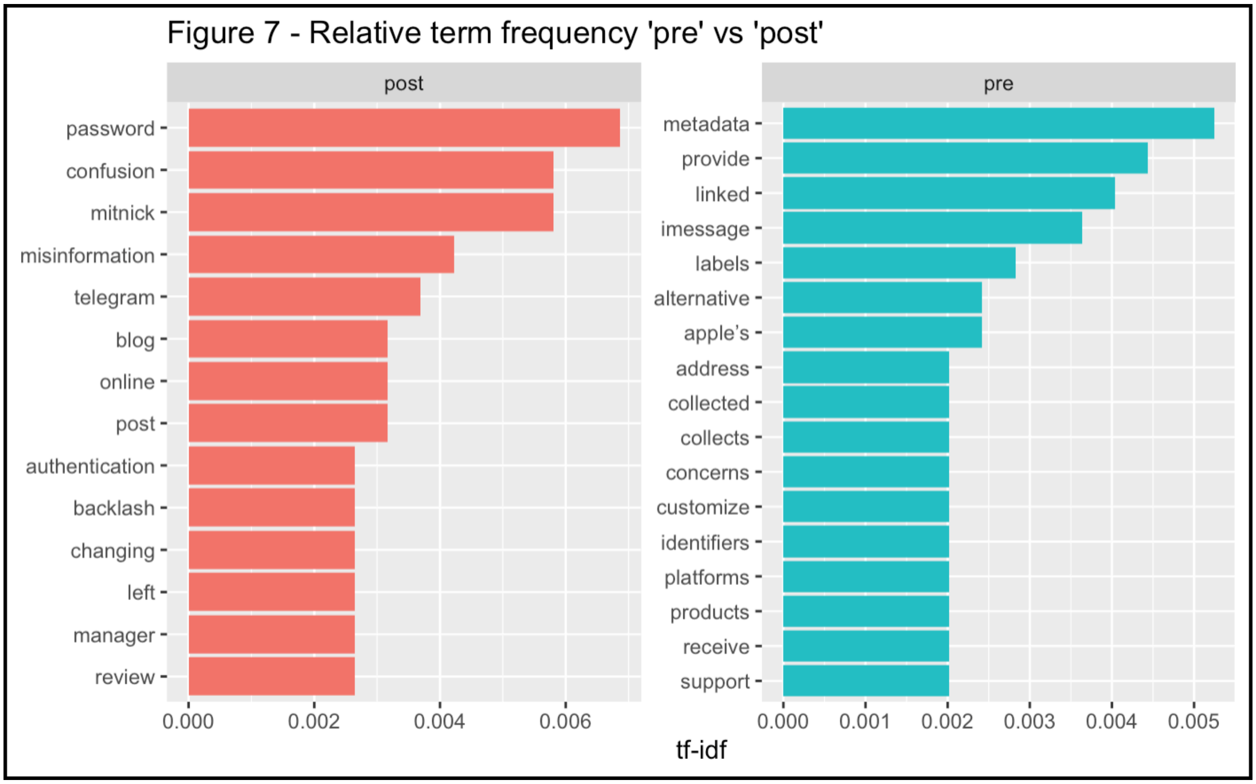


Figure 6 - Contribution to sentiment (afinn)

# Overall analysis

## Figure 7 – Term Frequency, Inverse Document Frequency

The relative most frequent terms found in the articles prior to any clarifying statement by WhatsApp suggest the articles focused on two topics. First, explaining what the privacy policy update was about, with terms such as ‘metadata’, ‘imessage’ and ‘provide’ having the highest tf-idf scores. Secondly, these articles provide insights into the cause behind the decision, changes made by ‘Apple’,  as well as ‘alternative’ services for customers to turn to. On the other hand, the post announcement articles are distinguished by their focus on reactions to the information update, highlighted by ‘confusion’ and ‘misinformation’.



## Figure 8 – Sentiments

If we first focus our attention on the words with negative connotations, we can easily determine the general sentiment expressed throughout the articles. Indeed, ‘worry/worried’ and ‘abuse’ have the largest impact on sentiment in the ‘pre’ articles, highlighting the concerns people had with the changes being made, as well as a feeling of being taken advantage of or an abuse of power on behalf of Facebook. In the ‘post’ articles, ‘misinformation’ contributes the most to the negative feelings in the articles. We cannot be sure whether this speaks to frustration over lack of clarity in WhatsApp’s first announcement, or the following spread of false information as speculations were made by various parties.

The terms classified as positive in the Afinn library provide less insights into the content of the articles as many can have double meaning or depend too much on the words around them to be interpreted on their own.

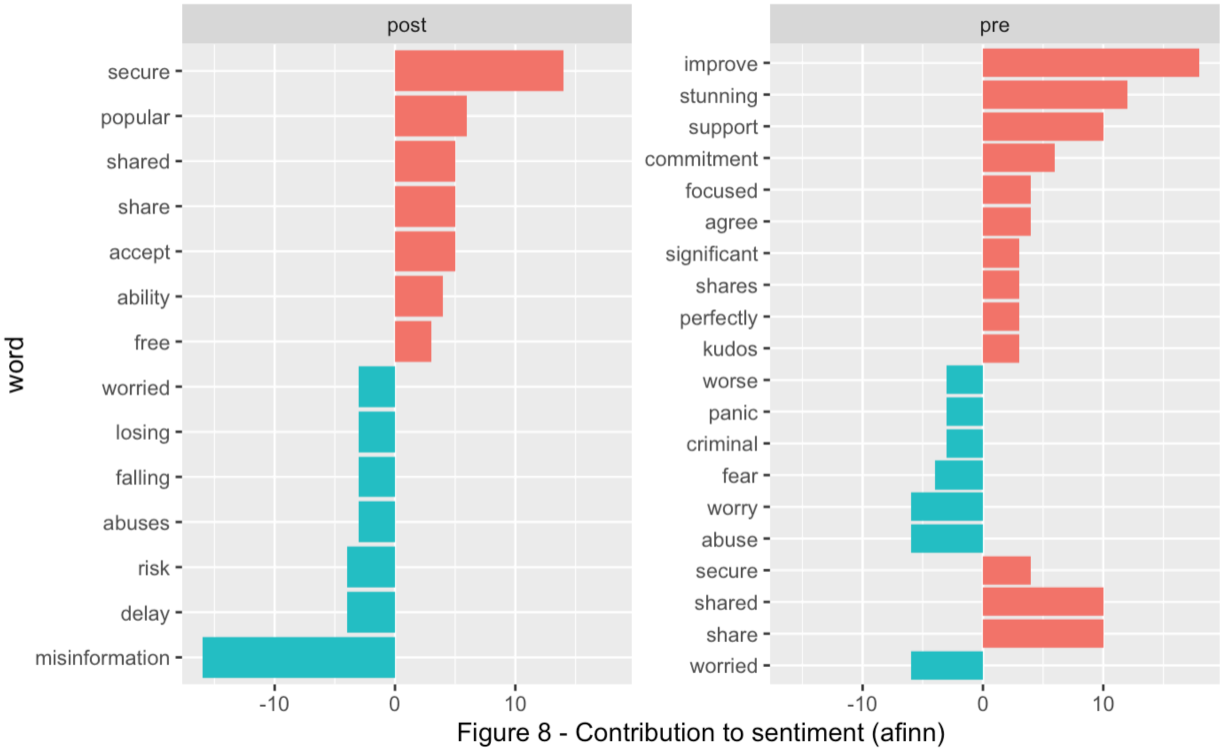


Figure 8 - Contribution to sentiment (afinn)

## Conclusion

To conclude, here are the top 3 insights our analysis generated:

- The initial statement by WhatsApp caused concern and fear in consumers' minds;
- The announcement prompted action, as articles put forth alternative options to WhatsApp's service;
- The correcting statement by WhatsApp had significant impact on the direction conversations were going, but did not lead to a positive outlook on the situation.

Information privacy has recently become a new hot topic, fueled by an increased awareness of the issue via documentaries such as The Social Dilemma. As such, and proven by WhatsApp's experience, companies need to be very careful with statements released to the public, specifically as they concern hot topics (like data privacy), of which consumers have varying degrees of understanding. While the second statement providing clarity into the update was the right move, it would not have been necessary had more thought been put into the first one. In other words, this displays the importance of asking outside parties to look over material, checking whether the message intended by the author, obviously very close to the subject, is properly understood and interpreted by someone seeing this information for the first time.

## References

Doffman, Z. (2021, January 3). *WhatsApp Beaten By Apple's New iMessage Privacy Update*. Forbes. https://www.forbes.com/sites/zakdoffman/2021/01/03/whatsapp-beaten-by-apples-new-imessage-update-for-iphone-users/

Griffin, A. (2021, January 9). *WhatsApp new privacy terms: What do new rules really mean for you?* The Independent. https://www.independent.co.uk/life-style/gadgets-and-tech/whatsapp-new-privacy-terms-facebook-rules-explained-b1784469.html

Hepworth, S. (2021, January 15). *Should you keep using WhatsApp? Plus five tips to start the year with your digital privacy intact*. The Guardian. https://www.theguardian.com/technology/2021/jan/16/should-you-keep-using-whatsapp-plus-five-tips-to-start-the-year-with-your-digital-privacy-intact

Hern, A. (2021, January 11). *Now WhatsApp users are really Facebook customers now – it's getting harder to forget that*. The Guardian. http://www.theguardian.com/commentisfree/2021/jan/11/whatsapp-facebook-app-privacy-policy

Isaac, M. (2021, January 15). *WhatsApp Delays Privacy Changes Amid User Backlash*. NYTimes. https://www.nytimes.com/2021/01/15/technology/whatsapp-privacy-changes-delayed.html

Kharpal, A. (2021, January 18). *WhatsApp delays privacy update over user "confusion" and backlash about Facebook data sharing*. CNBC. https://www.cnbc.com/2021/01/18/whatsapp-delays-privacy-update-amid-facebook-data-sharing-confusion.html

Newman, L. H. (2021, January 8). WhatsApp Has Shared Your Data With Facebook for Years, Actually. *Wired*. https://www.wired.com/story/whatsapp-facebook-data-share-notification/

Statt, N. (2021, January 15). *WhatsApp to delay new privacy policy amid mass confusion about Facebook data sharing*. The Verge. https://www.theverge.com/2021/1/15/22233257/whatsapp-privacy-policy-update-delayed-three-months

# Appendix - WhatsApp's privacy policy update

Elise Neumann

2/10/2021

## Loading all data and libraries

```
library(textreadr)
library(dplyr)
library(tidytext)
library(tidyverse)
library(ggplot2)
library(scales)
library(tm)
library(ggraph)
library(igraph)

data(stop_words)
```

```
## for articles pre whatsapp clarification
# Importing all .txt files from one directory
setwd("/Users/anneliseneumann/Desktop/Text Analytics and NLP/Assignement 1/txt files whatsapp pre")
nm <- list.files(path=
    "/Users/anneliseneumann/Desktop/Text Analytics and NLP/Assignement 1/txt files whatsapp pre")


# using read document to import the data:
my_txt_text <- do.call(rbind, lapply(nm, function(x) paste(read_document(file=x),
                                                            collapse = " ")))


# restructuring data as a data frame
mydf_pre <- data_frame(release = "pre", location=c("Forbes", "Guardian", "Independent",
            "Wired"), text=my_txt_text)  # putting in location info and text col
```

```
## Warning: ‘data_frame()‘ is deprecated as of tibble 1.1.0.
## Please use ‘tibble()‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_warnings()‘ to see where this warning was generated.
```

```
# creating a tidy version of the df
tidy_pre <- mydf_pre %>%
  unnest_tokens(word,text) %>%
  anti_join(stop_words) %>%
```

```r
  count(location, word, sort=TRUE) %>%
  mutate(word=reorder(word, n)) %>%
  group_by(location)


## for articles post whatsapp clarification
# Importing all .txt files from one directory
setwd("/Users/anneliseneumann/Desktop/Text Analytics and NLP/Assignement 1/txt files whatsapp")
nm <- list.files(path=
    "/Users/anneliseneumann/Desktop/Text Analytics and NLP/Assignement 1/txt files whatsapp")


# using read document to import the data:
my_txt_text <- do.call(rbind, lapply(nm, function(x) paste(read_document(file=x),
                                                           collapse = " ")))


# restructuring data as a data frame
mydf_post <- data_frame(release = "post", location=c("CNBC", "Guardian", "NYT", "Verge"),
                        text=my_txt_text)  # putting in location info and text col


# creating a tidy version of the df
tidy_post <- mydf_post %>%
  unnest_tokens(word,text) %>%
  anti_join(stop_words) %>%
  count(location, word, sort=TRUE) %>%
  mutate(word=reorder(word, n)) %>%
  group_by(location)


## Consolidated
mydf <- rbind(mydf_pre, mydf_post)

# creating a tidy version of the df
tidy_df <- mydf %>%
  unnest_tokens(word,text) %>%
  anti_join(stop_words) %>%
  count(release, word, sort=TRUE) %>%
  mutate(word=reorder(word, n)) %>%
  group_by(release)
```

## Analysis of pre articles

### Relative term frequency with tf_idf

```r
# calculating total words by article
total_words <- tidy_pre %>%
  group_by(location) %>%
  summarize(total = sum(n))
```

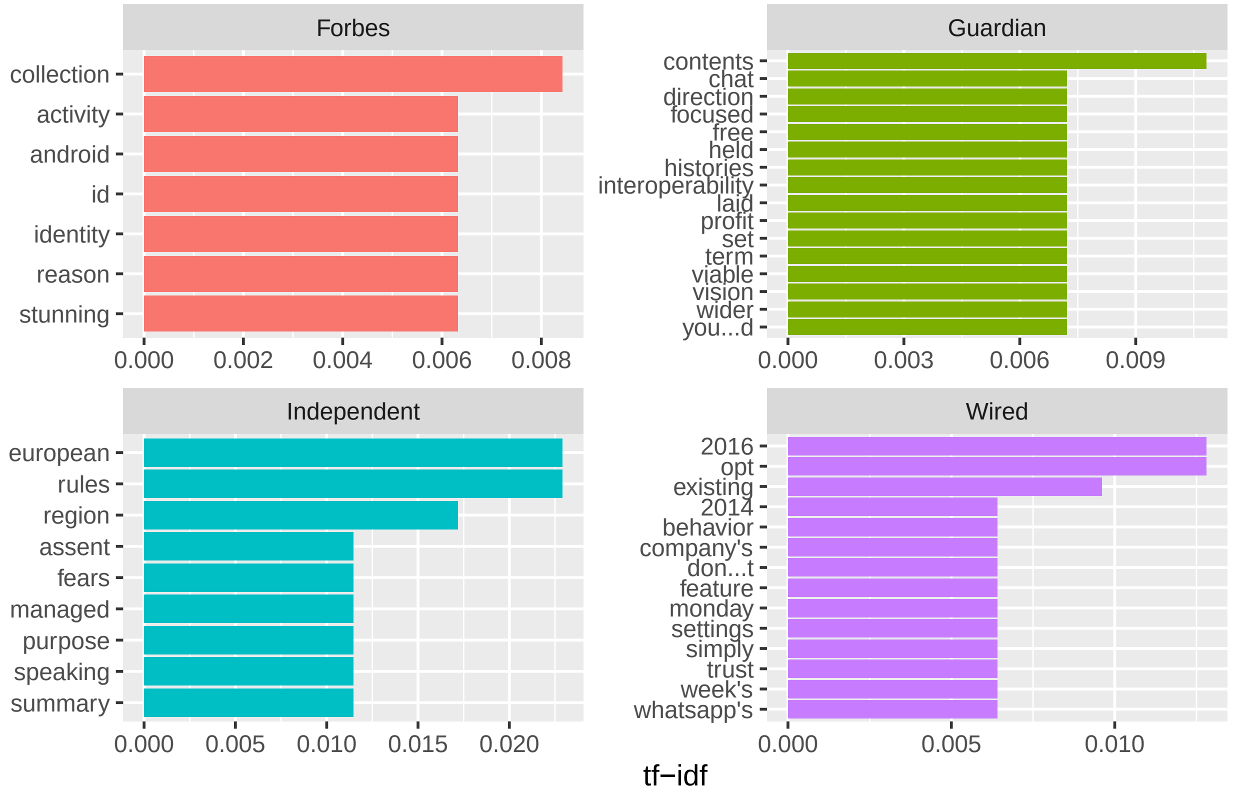
```
total_words <- left_join(tidy_pre, total_words)
```

```
## Joining, by = "location"
```

```
## Visualizing distribution
# ggplot(total_words, aes(n/total, fill = location)) +
#   geom_histogram(show.legend = FALSE) +
#   facet_wrap(~location, ncol = 2, scales = "free_y")


# table ordered by tf_idf scores
article_words <- tidy_pre %>%
  bind_tf_idf(word, location, n) %>%
  arrange(desc(tf_idf)) %>%
  filter(n<5)


# # Zipf’s law
# total_words %>%
#   group_by(location) %>%
#   mutate(rank = row_number(), ‘term frequency‘ = n/total) %>%
#   ggplot(aes(rank, ‘term frequency‘, color = location)) +
#     geom_abline(intercept = -0.62, slope = -1.11, color = "gray50", linetype = 2) +
#     geom_line(size = 1.1, alpha = 0.8, show.legend = FALSE) +
#     scale_x_log10() +
#     scale_y_log10()


# graphical approach
article_words %>%
  arrange(desc(tf_idf)) %>%
  mutate(word=factor(word, levels=rev(unique(word)))) %>%
  group_by(location) %>%
  filter(n<5) %>%
  top_n(5) %>%
  ungroup() %>%
  ggplot(aes(word, tf_idf, fill=location))+
    geom_col(show.legend=FALSE)+
    labs(title = "Figure 1 - Relative term frequency for ’pre’ articles",
         x=NULL, y="tf-idf")+
    facet_wrap(~location, ncol=2, scales="free")+
    coord_flip()
```

```
## Selecting by tf_idf
```

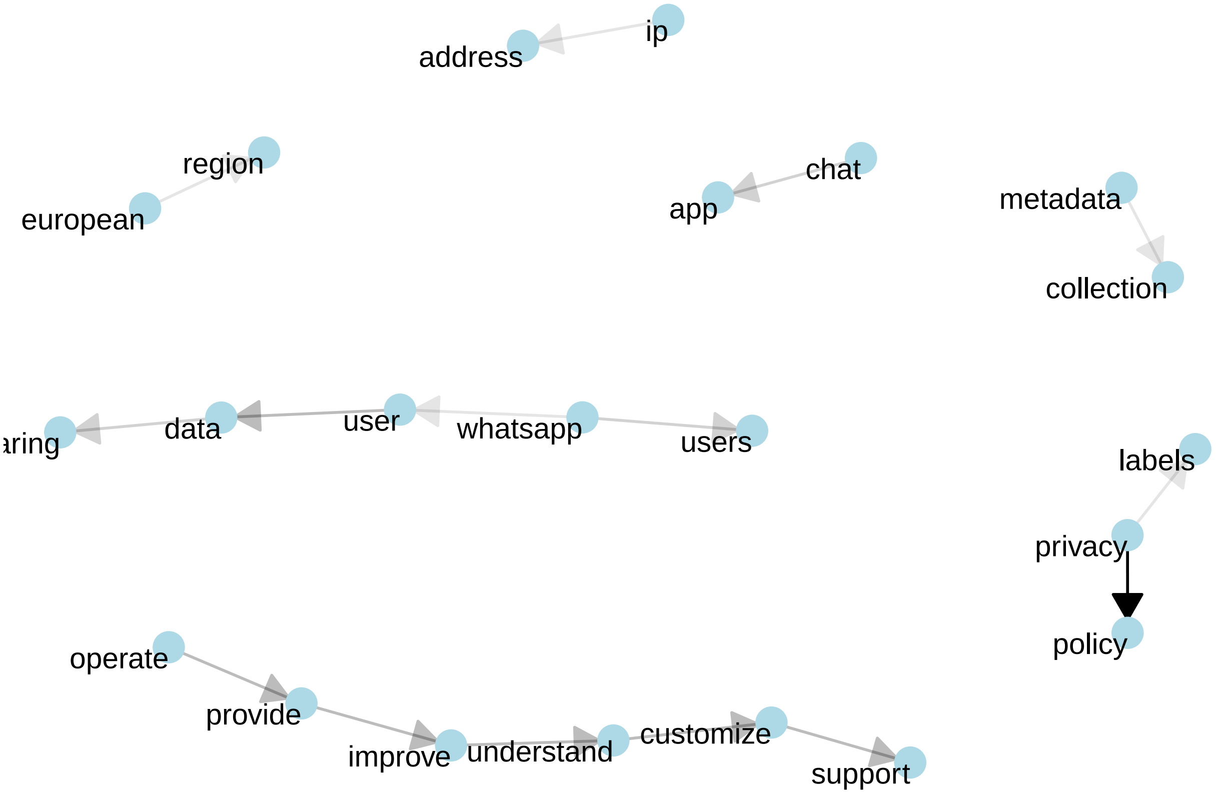
Figure 1 − Relative term frequency for 'pre' articles



### Bigrams for insights into deeper meaning in phrases

```
# creating bigrams
bigrams <- mydf_pre %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  filter(!is.na(bigram)) %>%
  separate(bigram, c("word1", "word2"), sep=" ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)


# visualizing links between words
set.seed(2016)
a <- grid::arrow(type = "closed", length = unit(.15, "inches"))

bigrams %>%
  filter(n>2) %>%
  graph_from_data_frame() %>%
  ggraph(layout = "fr") +
  geom_edge_link(aes(edge_alpha = n), show.legend = FALSE,
                 arrow = a, end_cap = circle(.07, ’inches’)) +
  geom_node_point(color = "lightblue", size = 5) +
  geom_node_text(aes(label = name), vjust = 1, hjust = 1) +
  theme_void() +
  labs(title = "Figure 2 - Bigram network for ’pre’ articles")
```

Figure 2 − Bigram network for 'pre' articles



## Sentiment analysis

```
# Visualizing contribution to sentiment with Afinn
tidy_pre %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(location) %>%
  mutate(n = n*value) %>%
  filter(abs(n) > 1) %>%
  mutate(word = fct_reorder(word, n)) %>%
  arrange(desc(n)) %>%
  ggplot(aes(word, n, fill = ifelse(n<0, "red", "green"))) +
    geom_bar(stat = "identity", show.legend = F) +
    facet_wrap(~location, scales = "free_y") +
    ylab("Figure 3 - Contribution to sentiment (afinn)") +
    coord_flip()
```

```
## Joining, by = "word"
```

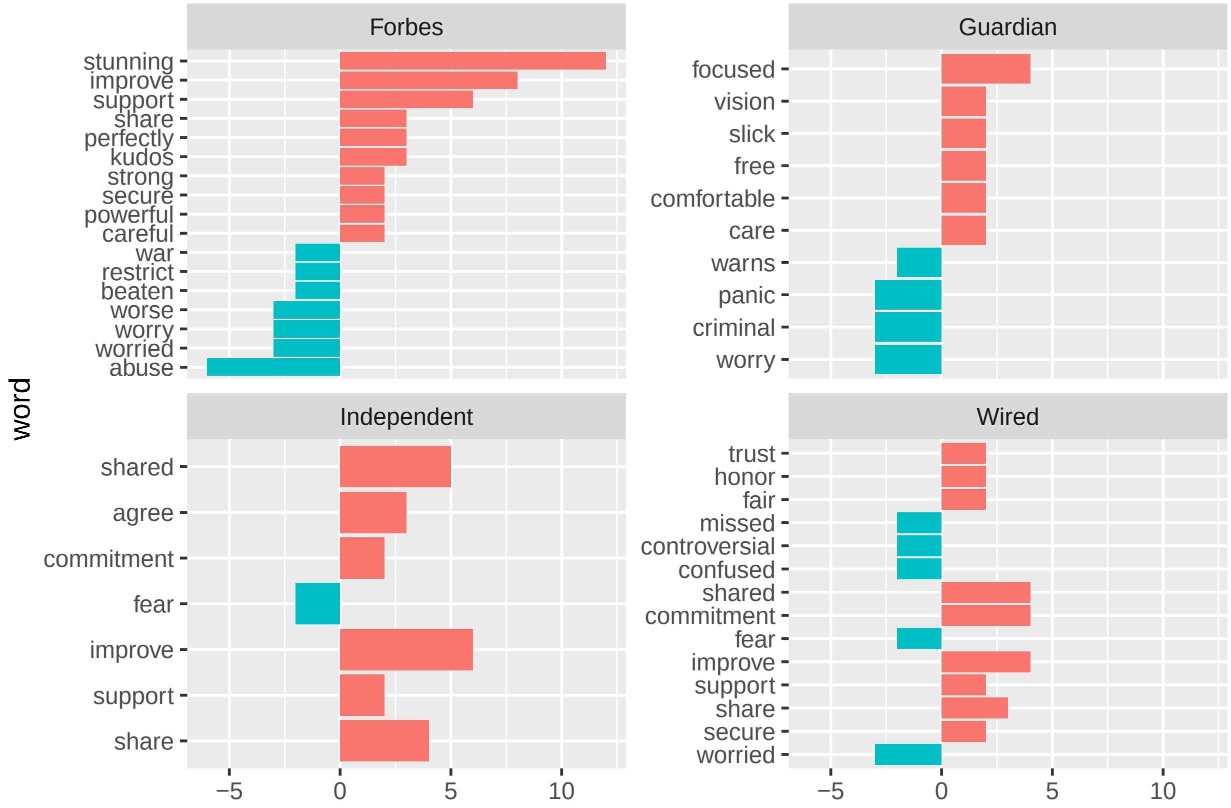


Figure 3 − Contribution to sentiment (afinn)

# Analysis of post articles

## Relative term frequency with tf_idf

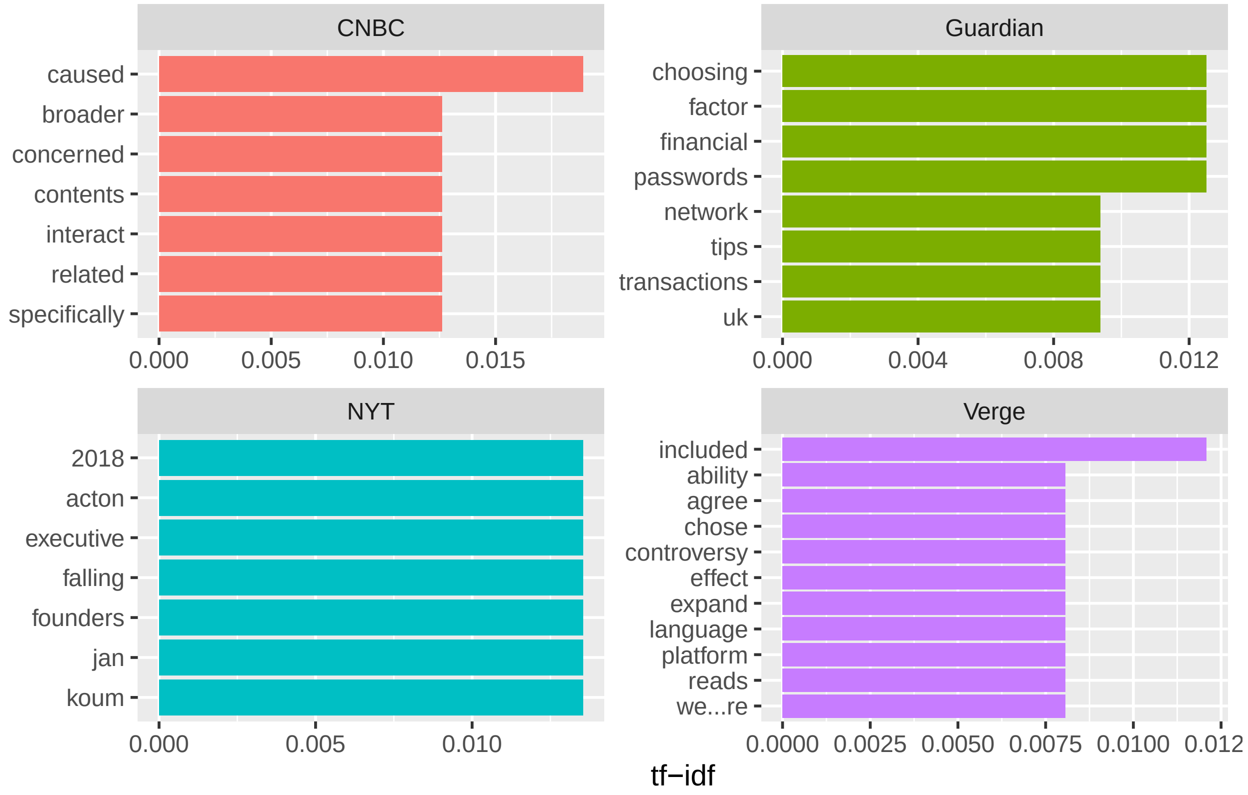
```
# calculating total words by article
total_words <- tidy_post %>%
  group_by(location) %>%
  summarize(total = sum(n))

total_words <- left_join(tidy_post, total_words)
```

```
## Joining, by = "location"
```

```
## Visualizing distribution
# ggplot(total_words, aes(n/total, fill = location)) +
#   geom_histogram(show.legend = FALSE) +
#   facet_wrap(~location, ncol = 2, scales = "free_y")


# table ordered by tf_idf scores
article_words <- tidy_post %>%
  bind_tf_idf(word, location, n) %>%
  arrange(desc(tf_idf)) %>%
  filter(n<5)


# # Zipf's law
# total_words %>%
#   group_by(location) %>%
#   mutate(rank = row_number(), `term frequency` = n/total) %>%
#   ggplot(aes(rank, `term frequency`, color = location)) +
#     geom_abline(intercept = -0.62, slope = -1.11, color = "gray50", linetype = 2) +
#     geom_line(size = 1.1, alpha = 0.8, show.legend = FALSE) +
#     scale_x_log10() +
#     scale_y_log10()


# graphical approach
article_words %>%
  arrange(desc(tf_idf)) %>%
  mutate(word=factor(word, levels=rev(unique(word)))) %>%
  group_by(location) %>%
  filter(n<5) %>%
  top_n(5) %>%
  ungroup() %>%
  ggplot(aes(word, tf_idf, fill=location))+
    geom_col(show.legend=FALSE)+
    labs(title = "Figure 4 - Relative term frequency for 'post' articles",
         x=NULL, y="tf-idf")+
    facet_wrap(~location, ncol=2, scales="free")+
    coord_flip()
```

```
## Selecting by tf_idf
```

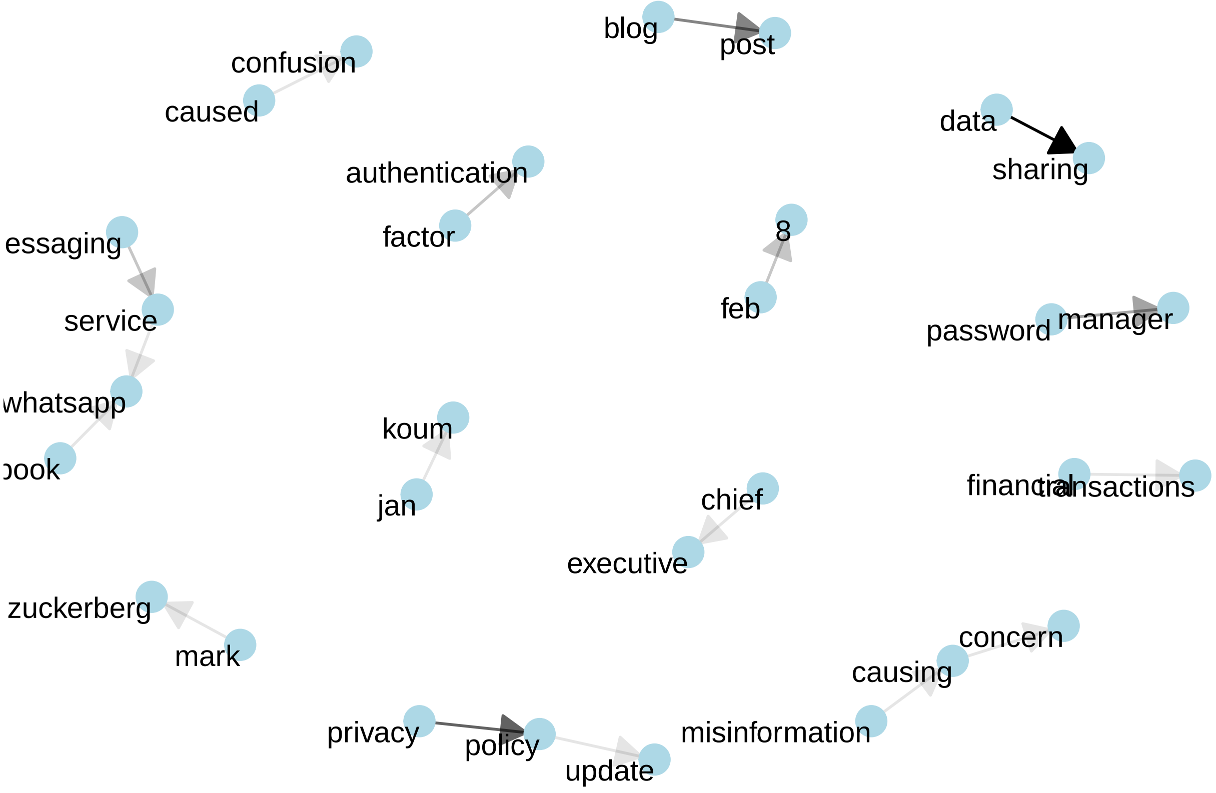
Figure 4 − Relative term frequency for 'post' articles



### Bigrams for insights into deeper meaning in phrases

```r
# creating bigrams
bigrams <- mydf_post %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  filter(!is.na(bigram)) %>%
  separate(bigram, c("word1", "word2"), sep=" ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)


# visualizing links between words
set.seed(2016)
a <- grid::arrow(type = "closed", length = unit(.15, "inches"))

bigrams %>%
  filter(n>2) %>%
  graph_from_data_frame() %>%
  ggraph(layout = "fr") +
  geom_edge_link(aes(edge_alpha = n), show.legend = FALSE,
                 arrow = a, end_cap = circle(.07, 'inches')) +
  geom_node_point(color = "lightblue", size = 5) +
  geom_node_text(aes(label = name), vjust = 1, hjust = 1) +
  theme_void() +
  labs(title = "Figure 5 - Bigram network for ’post’ articles")
```

Figure 5 − Bigram network for 'post' articles



### Sentiment analysis

```
# Visualizing contribution to sentiment with Afinn
tidy_post %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(location) %>%
  mutate(n = n*value) %>%
  filter(abs(n) > 1) %>%
  mutate(word = fct_reorder(word, n)) %>%
  arrange(desc(n)) %>%
  ggplot(aes(word, n, fill = ifelse(n<0, "red", "green"))) +
    geom_bar(stat = "identity", show.legend = F) +
    facet_wrap(~location, scales = "free_y") +
    ylab("Figure 6 - Contribution to sentiment (afinn)") +
    coord_flip()
```

```
## Joining, by = "word"
```

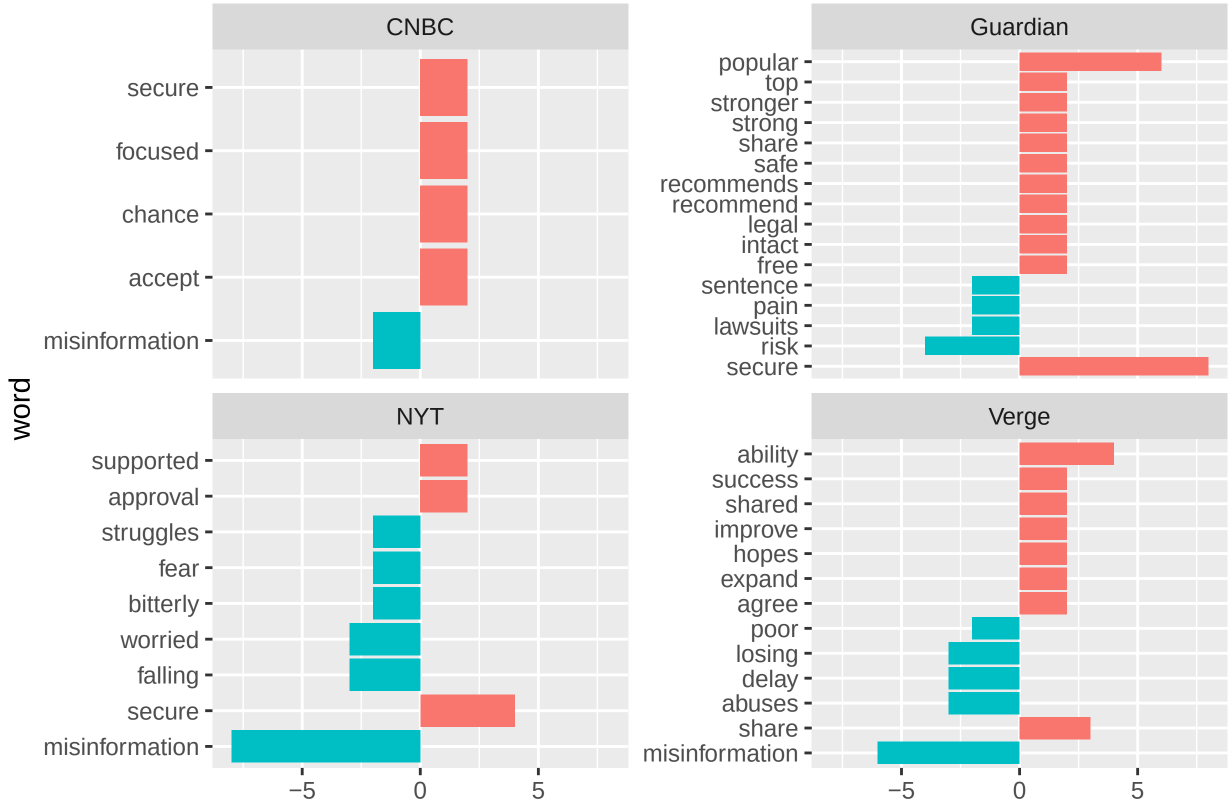


Figure 6 − Contribution to sentiment (afinn)

# Overall analysis

## Relative term frequency with tf_idf

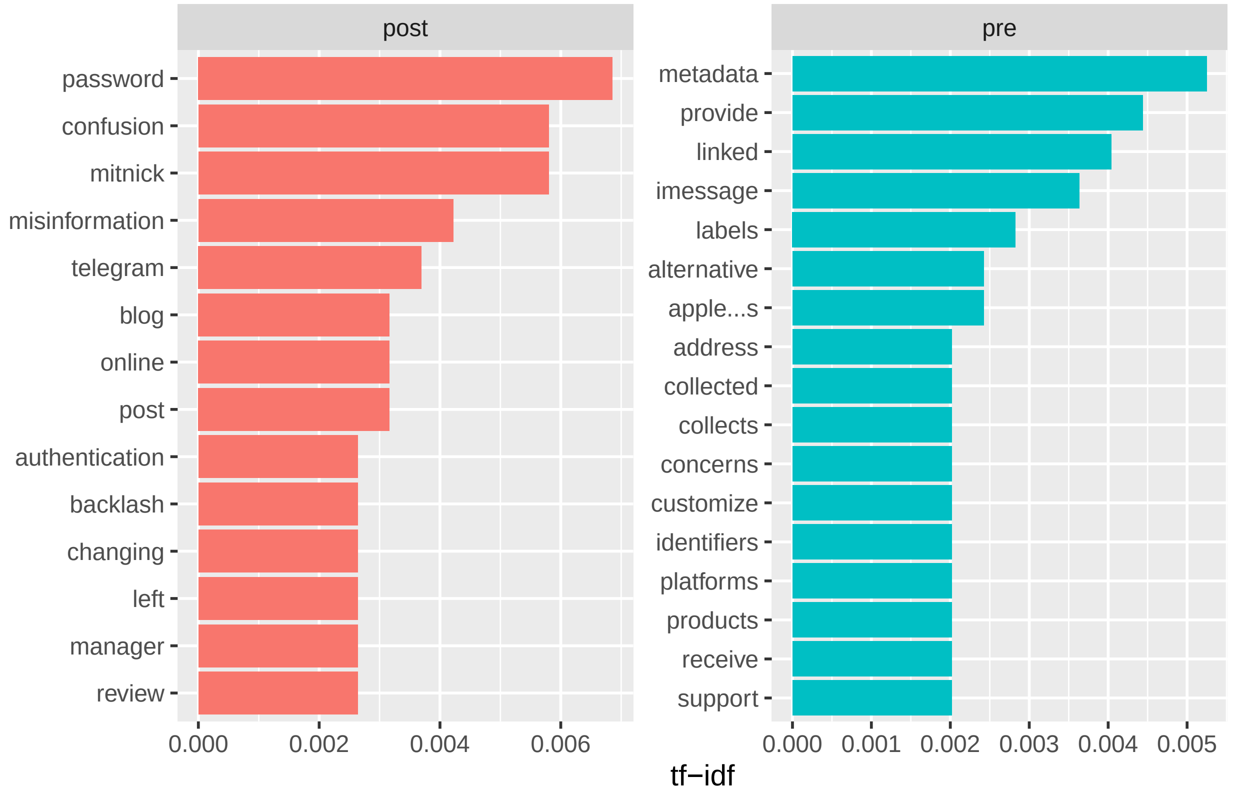
```
# calculating total words by article
total_words <- tidy_df %>%
  group_by(release) %>%
  summarize(total = sum(n))

total_words <- left_join(tidy_df, total_words)
```

```
## Joining, by = "release"
```

```
## Visualizing distribution
# ggplot(total_words, aes(n/total, fill = release)) +
#   geom_histogram(show.legend = FALSE) +
#   facet_wrap(~release, ncol = 2, scales = "free_y")


# table ordered by tf_idf scores
article_words <- tidy_df %>%
  bind_tf_idf(word, release, n) %>%
  arrange(desc(tf_idf))


# # Zipf’s law
# total_words %>%
#   group_by(release) %>%
#   mutate(rank = row_number(), ‘term frequency‘ = n/total) %>%
#   ggplot(aes(rank, ‘term frequency‘, color = release)) +
#     geom_abline(intercept = -0.62, slope = -1.11, color = "gray50", linetype = 2) +
#     geom_line(size = 1.1, alpha = 0.8, show.legend = FALSE) +
#     scale_x_log10() +
#     scale_y_log10()


# graphical approach
article_words %>%
  arrange(desc(tf_idf)) %>%
  mutate(word=factor(word, levels=rev(unique(word)))) %>%
  group_by(release) %>%
  top_n(10) %>%
  ungroup() %>%
  ggplot(aes(word, tf_idf, fill=release))+
    geom_col(show.legend=FALSE)+
    labs(title = "Figure 7 - Relative term frequency ’pre’ vs ’post’",
         x=NULL, y="tf-idf")+
    facet_wrap(~release, ncol=2, scales="free")+
    coord_flip()
```

```
## Selecting by tf_idf
```

Figure 7 − Relative term frequency 'pre' vs 'post'



## Sentiment analysis

```
# Visualizing contribution to sentiment with Afinn
tidy_df %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(release) %>%
  mutate(n = n*value) %>%
  filter(abs(n) > 2) %>%
  mutate(word = fct_reorder(word, n)) %>%
  arrange(desc(n)) %>%
  ggplot(aes(word, n, fill = ifelse(n<0, "red", "green"))) +
    geom_bar(stat = "identity", show.legend = F) +
    facet_wrap(~release, scales = "free_y") +
    ylab("Figure 8 - Contribution to sentiment (afinn)") +
    coord_flip()
```

```
## Joining, by = "word"
```

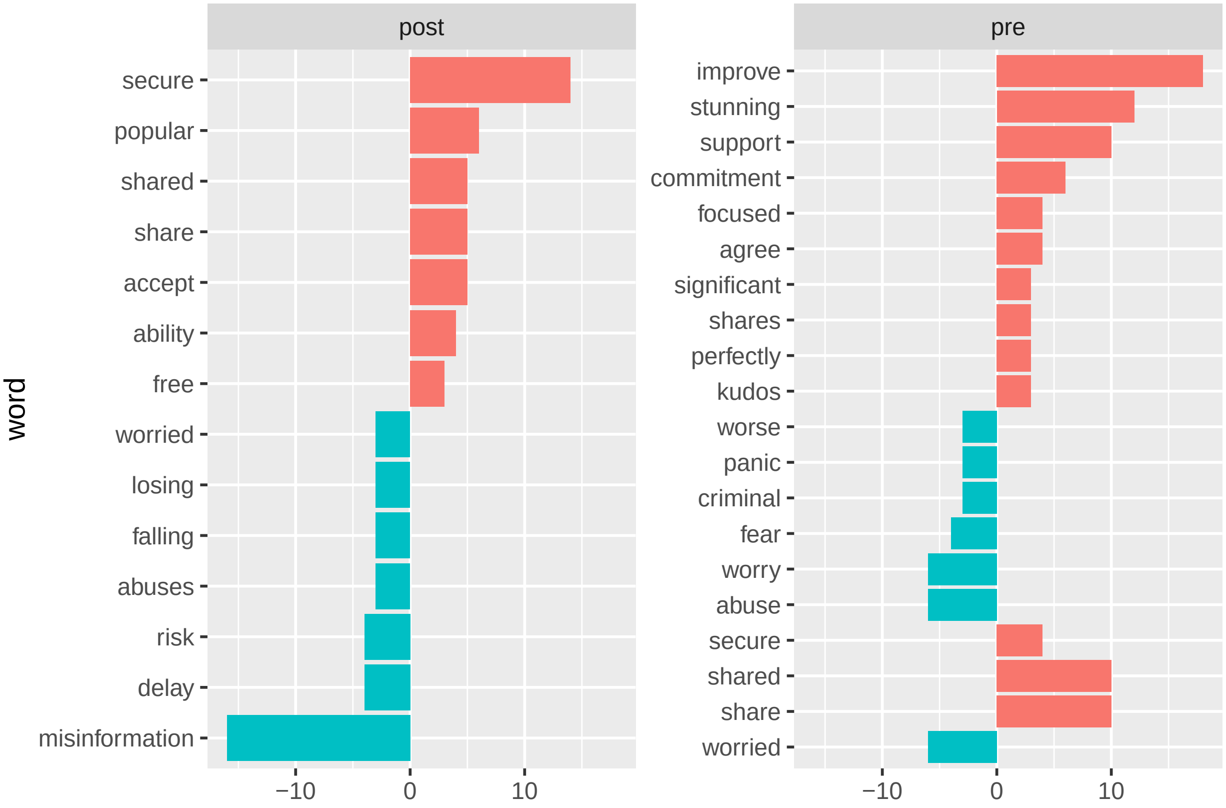


Figure 8 − Contribution to sentiment (afinn)