Text Analytics and Natural Language Processing – DAT-5317

A3: Business Insights Report

Bitcoin @ Twitter: Marketing “BitCover” ETF

By

Marius Heje Maehle

05. December 2021

Bitcoin @ Twitter: Marketing BitCover ETF

Business case: A new Bitcoin Exchanged Traded Fund (ETF) called "BitCover" want to market their fund on Twitter. Which words and sentiments should they focus on in their marketing on Twitter to acquire customers?

Bitcoin and Twitter

Twitter as a Social Media platform

As twitter has been one of the most used social media platforms the last years (Most popular social media networks, 2021) it can provide a lot of data and value to consumer. As twitter provides a lot of text it can be used to find valuable insights, but also there can be a lot of noise in the data. All the data that is retrieved and used in the analysis is from Twitter's database.

Cryptocurrencies and Bitcoin

Since the pandemic there has been a lot of new investors coming into the market due to a post-pandemic boom (Reinicke, 2021). I will not go into depth why this is, but it is important to acknowledge as the stock market and cryptocurrencies has been booming since the crash in the spring 2020 (Reinicke 2021).

As seen on the chart below the total cryptocurrency market cap has overall increased a lot towards almost a market cap now around 2,5 trillion dollars (Global Cryptocurrency Charts, 2021). The cryptocurrency has a history of being volatile, but the market size has increased exponentially since 2020 (Global Cryptocurrency Charts, 2021).

Chart 1: Total cryptocurrency Market Cap. Retreived from: Global Cryptocurrency Charts, 2021

<u>BitCover ETF</u>

As there are more investors in the market, there is a lot of hype and interest around cryptocurrencies which has created a new wave of crypto traders as seen on the Global Cryptocurrency Chart (2021). A bitcoin ETF mimics the price of the digital currency to allow investor to buy into the ETF without trading bitcoin itself (Reiff, 2021). This can make it easier for investors to buy and own bitcoin on the market. To separate the signal from the noise in all the data on twitter these days can be difficult, and BitCover wants to se be signal in the data. As BitCover is a newly started ETF, they want to see how they should approach their words association and word sentiments in their marketing to acquire and get more customers.

The business problem statement is as mentioned: *"Which words and sentiments should they focus on in their marketing on Twitter to acquire customers?"*

Analysis of the Twitter data

The data analysis will focus on popular words, sentiments, positive and negative words, and word connections to give some valuable output from the data to provide useful business decision for the twitter marketing.

Sentiment analysis

As sentiment analysis provide a good way to understand the attitude and opinions expressed in the text it can give valuable insight on which word has positive sentiments regarding bitcoin (Silge & Robinson, 2017). NRC is the most important sentiment library as it offers a lot of different flavors of sentiments and gives the best structure to provide valuable business insights. As seen on the table below the NRC method is giving a positive sentiment, while the Bing and Afinn is both giving negative sentiments values.

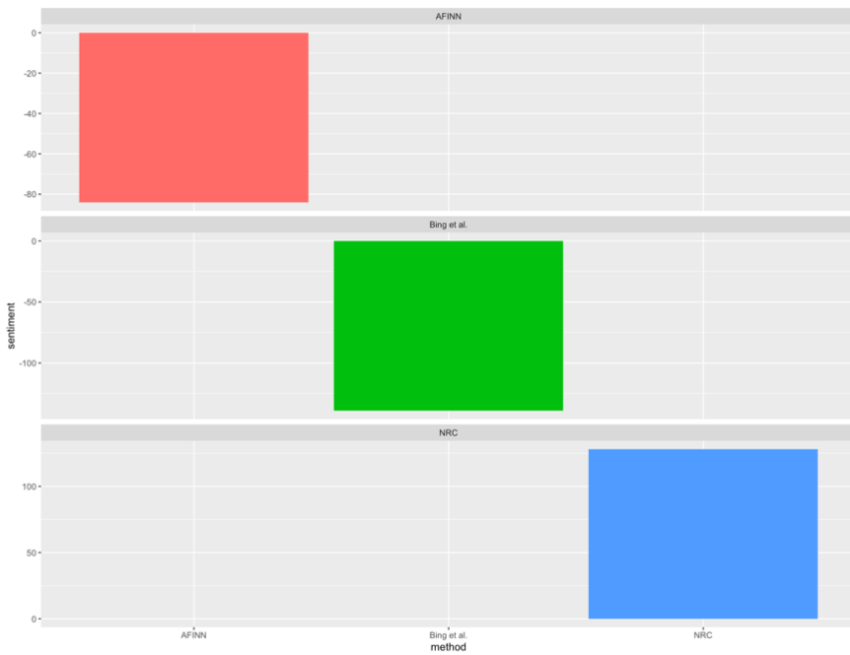


Chart 2: Sentiment methods

Most common positive and negative words

When looking at the most positive and negative words in the data it is important so which can be leveraged in a positive way in the marketing, and which words that should be avoided.

Positive words: As we clearly see that the positive words are strongly connected to feelings with **super**, **support**, **strong**, **love** and **happy**. These words should be used on how the marketing should be framed. The positive words such as **free**, **super**, **superb**, **bullish**, and **top** can be used more in the way to describe the investment strategy and the meaning of bitcoin. As bitcoin is seen by some as a hedge against other investments, I'm assuming that **gold** is used in the text to demonstrate the investment strategy of investing in bitcoin, but this is only an assumption.

Negative words: All the most common negative words are probably used when the people on twitter use bitcoin as a negative investment and currency. **Resistance** is maybe the only negative word which can be seen as positive in a way that bitcoin is seen as an investment. Still, it's good to know that the word can be seen as negative and should therefore be avoided in usage in the BitCover marketing.

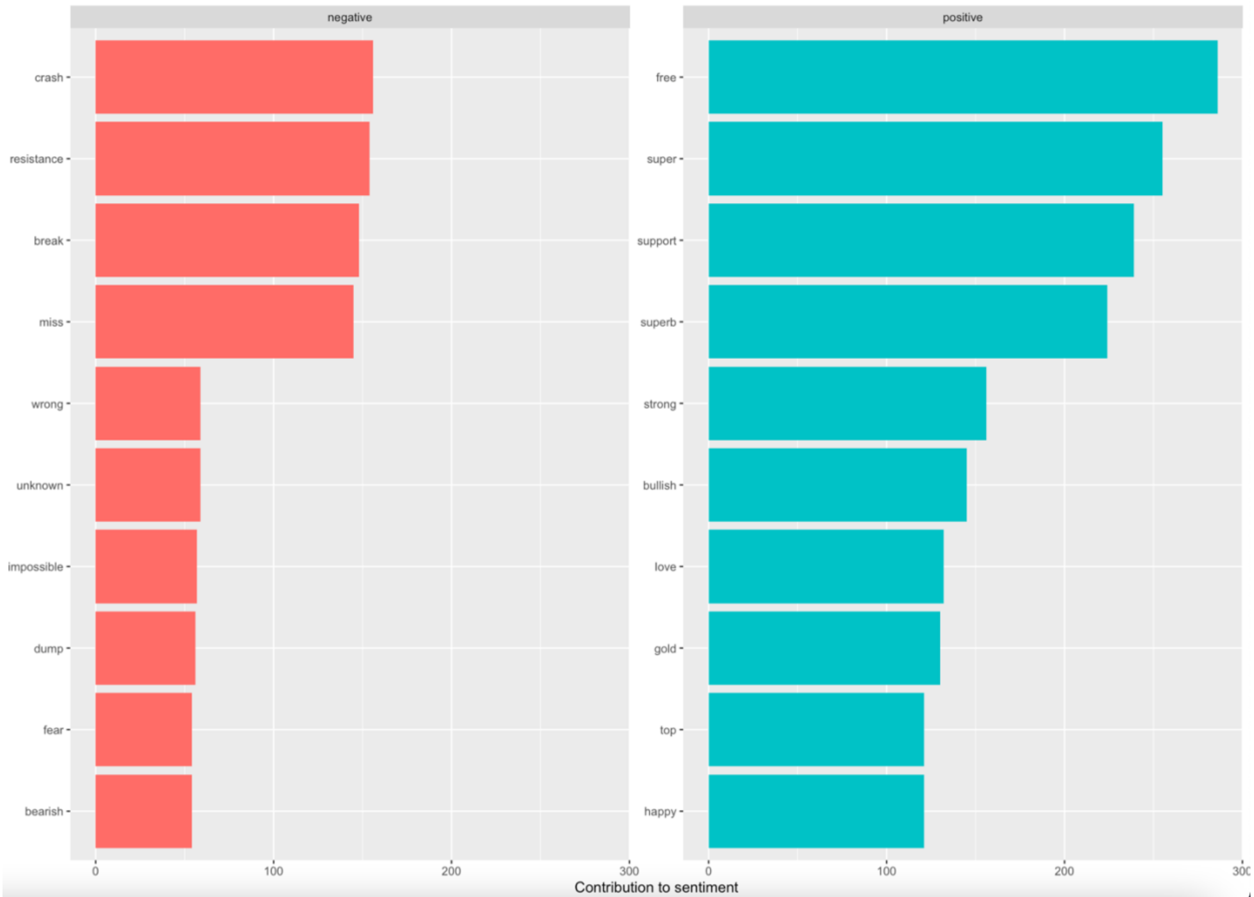


Chart 2.1: Most common positive and negative words

<u>N-grams</u>

By using the n-grams we can see which words tend to appear after others and how the relationship between the words is (Silge & Robinson, 2017). The arrow on the bigrams shows us the arrow to end before touching the node on which word follows which. We clearly see that bitcoin is connected to many words on the bigrams, but it is important see which word that can be useful to promote BitCover's ETF and filter out the noise. Words such as **crypto**,

cryptocurrencies, **blockchain**, **price**, and **pushes** can be useful words to know about. Most of them makes sense without the bigrams, but it’s important to highlight the words people connect in pairs in connection to Bitcoin. Connections such as bitcoin → **btc** → **transactions** highlight the important of using bitcoin as a transaction. **Bitcoin** → **crypto** → **gamify and bitcoin** → **nft** → **gamefi** on how people connect bitcoin to gamification and NFTs, and how this can be leveraged long-term. Lastly, **bitcoin** → **usd** → **prices** on how bitcoin is compared to the United States Dollar and price range.

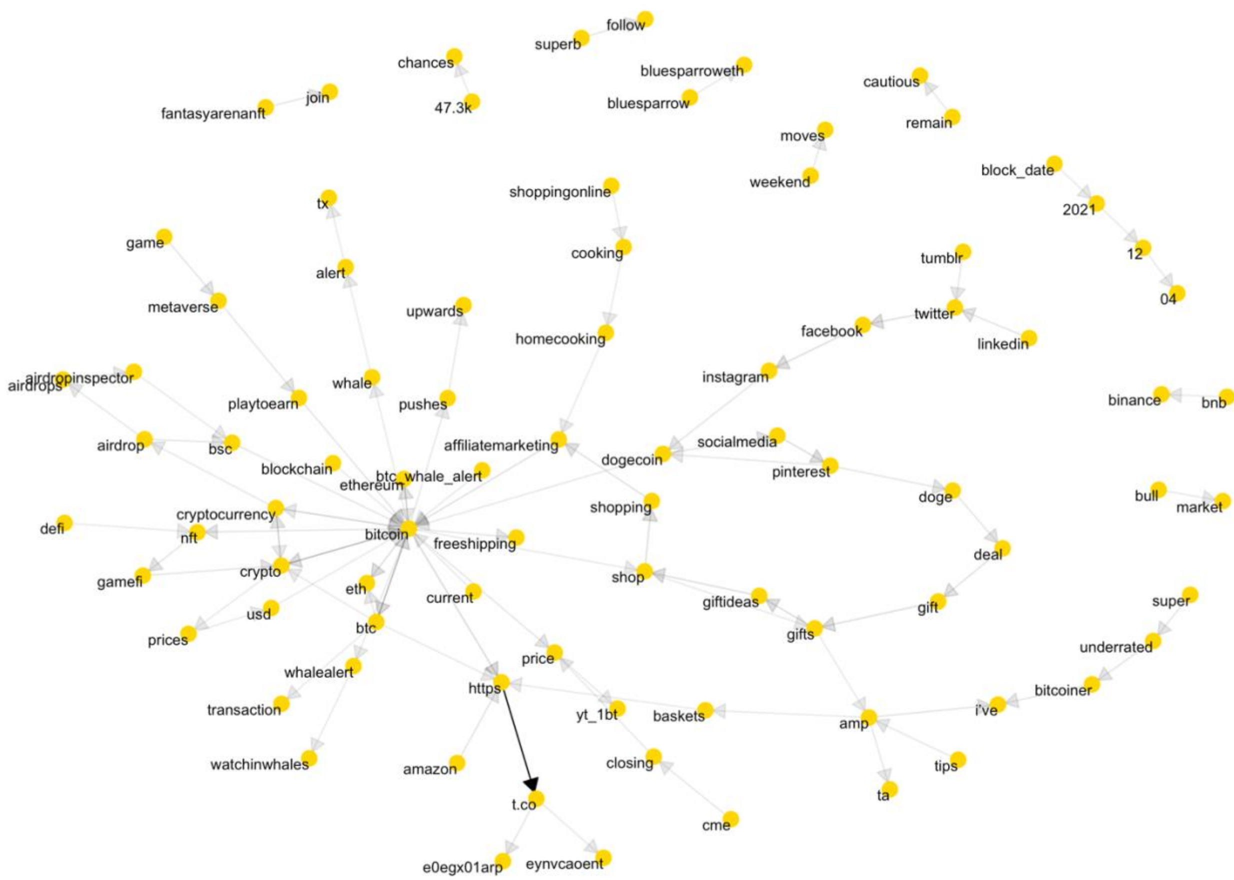


Chart 3: Common bigrams in the bitcoin twitter data

Pitfalls

It’s important to mention that the analysis should be carefully used because correlation does not always imply causation. The sample data from twitter gives a clue on what words which is popular, which ones are used together and which gives the different sentiments. If we want to investigate it further we should take more sample data, to find more connections and which ones that have a strong correlation between each other.

Business decision

How does a “normal” bitcoin trader look like?

Based on the facts from the background, sentiment analysis and n-grams we can see that there are some words and sentiments that should be focused on in the marketing of BitCover’s ETF. Keep it simple, and straight forward with playing on the sentiments of **support**, **strong**, **love** and **happy** to trigger the feelings of the buyer. In the more concrete wording in relation to bitcoin strategy use words such as **super**, **bullish**, and **top**. To provide more depth in relation to the word we found in the n-grams by connecting it to **crypto** and **blockchain**. More concrete BitCover can market their ETF that it is easy to focus on **price** in relation to the **USD**, and the easier way of doing **transaction** with bitcoin and BitCover’s ETF.

As these insights are straight forward and not so in depth, BitCover should further look deeper into the words and correlation between the words to get a better understanding which words helps create customers buying their Bitcoin ETF. By monitoring the change in words used over time can also be useful to change with the tides, such as the use of **NFTs** and **gamify**.

Appendix:

Global Cryptocurrency Charts (05.12.2021). *CoinMarketCap*. Retrieved from: https://coinmarketcap.com/charts/

Most popular social media networks (updated for 2021) – digital marketing's most powerful tool. (26.08.2021). *Revive.digital.* Retrieved from: https://revive.digital/blog/most-popular-social-media/

Reiff, N. (19.11.2021). Bitcoin ETFs Explained. *Investorpedia*. Retrieved from: https://www.investopedia.com/investing/bitcoin-etfs-explained/

Reinicke, C. (02.08.2021). New investors are jumping into the market during the post-pandemic boom. *CNBC*. Retrieved from: https://www.cnbc.com/2021/08/02/new-investors-are-jumping-into-the-market-in-the-post-pandemic-boom-.html

Silge, J. & Robinson, D. (2017). Text Mining with R: A Tidy Approach. *O'Reilly Media*

R Code and R output:

```
####################################################
######A3: Business Insights Report - Marius Heje Mæhle #######
###############Querying Twitter for bitcoins################

#install the necessary packages
#install.packages("twitteR")
#install.packages("tm")
#install.packages("NLP")
#install.packages("rtweet")

library(rtweet)
library(twitteR)
library(tm)
library(NLP)

#consumerKey, consumerSecret, access token and access secret from my twitter
consumer_key <- 'osmot3QLGT5ithOP2A3CgHGHh'
consumer_secret <- '2hP8tl70oJHeBLW0AZOaNfbZQj4hc51HeBFfkO310IV6CJleO6'
access_token <- '1204793374638837761-HXz9ePnCBLE93FJFk2mdet0rxQj2wc'
access_secret <- 'DpTIPGkY0H9PGHFVW3GSSrZYkOdC3oV3pBuPNyNVoxVBz'
name_of_app <- "Incerto Coding"

twitter_token <- create_token(
  app = name_of_app,
  consumer_key = consumer_key,
  consumer_secret = consumer_secret,
  access_token = access_token,
```

access_secret = access_secret)

rt <- search_tweets("#Bitcoin", n = 10000, lang = "en", include_rts = FALSE)

#I receive a Warning messages:

#1:In class(object) <- "environment" :Setting class(x) to "environment" sets attribute to NULL; result will no longer be an S4 object

#put dataset into a dataframe

library(dplyr)

mydf <- data.frame(line=1:9037, text=rt$text)

print(mydf)

token_list <- mydf %>%

unnest_tokens(word, text) %>%

count(word, sort=TRUE)

print(token_list)

```
> print(token_list)
              word     n
1          bitcoin 10595
2              the  5562
3             t.co  5363
4            https  5357
5               to  3766
6                a  3334
7               is  3102
8           crypto  2981
9              and  2816
10             btc  2716
```

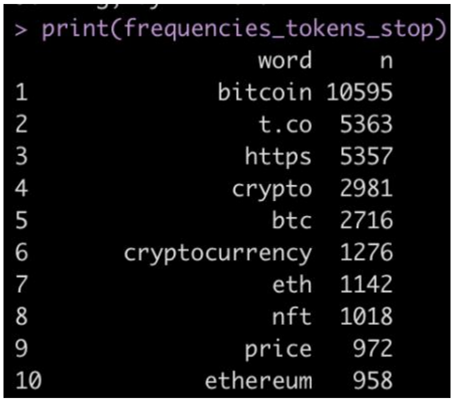
#remove stopwords

library(stringr)

data(stop_words)

frequencies_tokens_stop <- mydf %>%

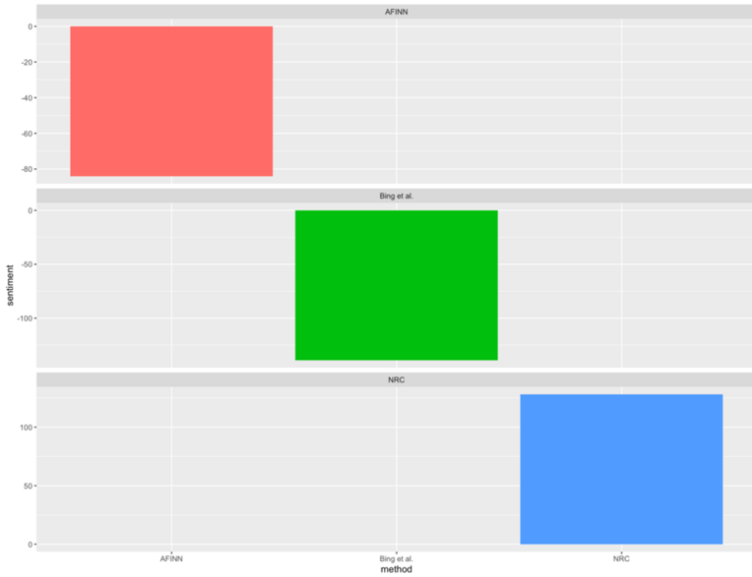
```
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  count(word, sort=TRUE)
print(frequencies_tokens_stop)
```



```
> print(frequencies_tokens_stop)
                word     n
1            bitcoin 10595
2               t.co  5363
3              https  5357
4             crypto  2981
5                btc  2716
6     cryptocurrency  1276
7                eth  1142
8                nft  1018
9              price   972
10          ethereum   958
```

```
### Sentiment analysis######
library(tidytext)
library(dplyr)
library(stringr)
library(tidyr)
library(tidytuesdayR)

nrcsurprise <- get_sentiments("nrc") %>%
  filter(sentiment == "surprise")

frequencies_tokens_stop %>%
  inner_join(nrcsurprise) %>%
  count(word, sort=T)

nrcsurprise <- get_sentiments("nrc") %>%
  filter(sentiment == "surprise")
```

```
frequencies_tokens_stop %>%
  inner_join(nrcsurprise) %>%
  count(word, sort=T)

afinn <- frequencies_tokens_stop %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

bing_and_nrc <- bind_rows(
  frequencies_tokens_stop%>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
  frequencies_tokens_stop %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)

library(ggplot2)
bind_rows(afinn, bing_and_nrc) %>%
  ggplot(aes(method, sentiment, fill=method))+
  geom_col(show.legend=FALSE)+
  facet_wrap(~method, ncol =1, scales= "free_y")
```



####### Most common positive and negative words #########

bing_counts <- frequencies_tokens_stop %>%

  inner_join(get_sentiments("bing")) %>%

  arrange(desc(n))

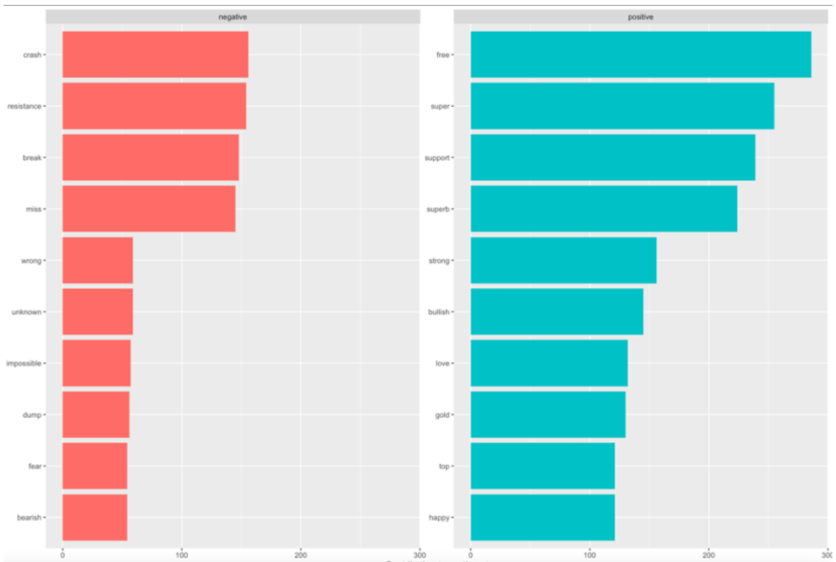
bing_counts

```
> bing_counts
             word   n sentiment
1            free 286  positive
2           super 255  positive
3         support 239  positive
4          superb 224  positive
5           crash 156  negative
6          strong 156  positive
7      resistance 154  negative
8           break 148  negative
9         bullish 145  positive
10           miss 145  negative
```

bing_counts %>%

  group_by(sentiment) %>%

```
  top_n(10, n) %>%
  ungroup() %>%
  mutate(word=reorder(word, n)) %>%
  ggplot(aes(word, n, fill=sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y")+
  labs(y="Contribution to sentiment", x=NULL)+
  coord_flip()
```




```
###### N-grams and tokenizing ###############
library(dplyr)
library(tidytext)
library(tidyr)
library(tidytuesdayR)

rt_bigrams_new <- rt %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2)

rt_bigrams_new #We want to see the bigrams (words that appear together, "pairs")
```

rt_bigrams_new %>%

  count(bigram, sort = TRUE) #this has many stop words, need to remove them

```
# A tibble: 84,754 × 2
   bigram               n
   <chr>            <int>
 1 https t.co        5354
 2 i’ve been          675
 3 crypto bitcoin     600
 4 is a               582
 5 btc bitcoin        568
 6 bitcoin btc        556
 7 bitcoin https      537
 8 to the             488
 9 gift gifts         442
10 giftideas shop     442
# … with 84,744 more rows
```

#remove stop words from the bigram data, use the separate function:

library(tidyr)

bigrams_separated <- rt_bigrams_new %>%

  separate(bigram, c("word1", "word2"), sep = " ")

#remove stopwords

bigrams_filtered <- bigrams_separated %>%

  filter(!word1 %in% stop_words$word) %>%

  filter(!word2 %in% stop_words$word)

#creating the new bigram, "no-stop-words":

bigram_counts <- bigrams_filtered %>%

  count(word1, word2, sort = TRUE)

#want to see the new bigrams

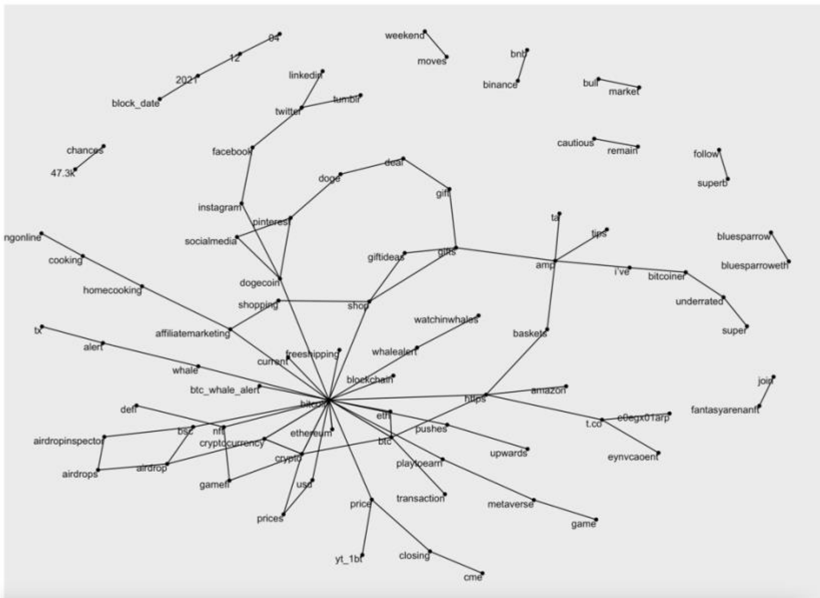
bigram_counts

```
> bigram_counts
# A tibble: 42,302 × 3
   word1     word2         n
   <chr>     <chr>     <int>
 1 https     t.co       5354
 2 crypto    bitcoin     600
 3 btc       bitcoin     568
 4 bitcoin   btc         556
 5 bitcoin   https       537
 6 gift      gifts       442
 7 giftideas shop        442
 8 gifts     giftideas   442
 9 twitter   facebook    442
10 facebook  instagram   441
# … with 42,292 more rows
```

```
###### VISUALISING THE BIGRAM NETWORK ##########

#install.packages("igraph")

library(igraph)

bigram_graph <- bigram_counts %>%
  filter(n>100) %>% #check the size, and edit proport
  graph_from_data_frame()


bigram_graph
```

```
> bigram_graph
IGRAPH 50f9d63 DN-- 86 96 --
+ attr: name (v/c), n (e/n)
+ edges from 50f9d63 (vertex names):
 [1] https            ->t.co          crypto          ->bitcoin
 [3] btc              ->bitcoin       bitcoin         ->btc
 [5] bitcoin          ->https         gift            ->gifts
 [7] giftideas        ->shop          gifts           ->giftideas
 [9] twitter          ->facebook      facebook        ->instagram
[11] shop             ->shopping      socialmedia     ->pinterest
[13] bitcoin          ->eth           bitcoin         ->crypto
[15] affiliatemarketing->bitcoin      deal            ->gift
+ ... omitted several edges
```

```
#install.packages("ggraph")

library(ggraph)

ggraph(bigram_graph, layout = "fr") +
  geom_edge_link()+
  geom_node_point()+
  geom_node_text(aes(label=name), vjust =1, hjust=1)
```




```
#polish the ngram to look better

a <- grid::arrow(type = "closed", length = unit(.15, "inches"))

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