Topic Modeling

Ada Lazuli

2022-07-28

```
library(tidymodels)
## -- Attaching packages -------------------------------------- tidymodels 1.0.0 --
## v broom        1.0.0     v recipes      1.0.1
## v dials        1.0.0     v rsample      1.0.0
## v dplyr        1.0.9     v tibble       3.1.8
## v ggplot2      3.3.6     v tidyr        1.2.0
## v infer        1.0.2     v tune         1.0.0
## v modeldata    1.0.0     v workflows    1.0.0
## v parsnip      1.0.0     v workflowsets 1.0.0
## v purrr        0.3.4     v yardstick    1.0.0
## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x purrr::discard() masks scales::discard()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x recipes::step()  masks stats::step()
## * Use tidymodels_prefer() to resolve common conflicts.
library(tidytext)
library(ggplot2)
library(reshape2)
##
## Attaching package: 'reshape2'
## The following object is masked from 'package:tidyr':
##
##     smiths
library(forcats)
library(topicmodels)
set.seed(101011)
```

Helper Functions

```
lda_pipeline <- function(dtm, k) {
  # execute the LDA algorithm
  return (LDA(dtm, k = k,method = "Gibbs",control = list(seed = 101011)) %>%
  # reshape the matrix to be in better form
  tidy(matrix = "beta") %>%
  # Group by topic
  group_by(topic) %>%
  # Take the top ten words with the
  top_n(10, beta) %>%
```

```
  # Un group the filtered dataframe
  ungroup() %>%
  # Create term2, a factor ordered by word probability
  mutate(ordered_term = fct_reorder(term, beta))
  )
}
```

Data Loading

```
df <- read.csv("../data/tweets_prepped.csv")
df$id <- 1:nrow(df)
```

Create Document Term Matrix

```
dtm <- df %>% unnest_tokens(word, tweet) %>%
  count(word, id) %>% cast_dtm(id, word, n) %>%
  as.matrix()
```

The `dtm` matrix is a sparse *document term* matrix has **8560** rows and **13181** columns.

Latent Dirichlet Allocation (LDA) Topic Modeling

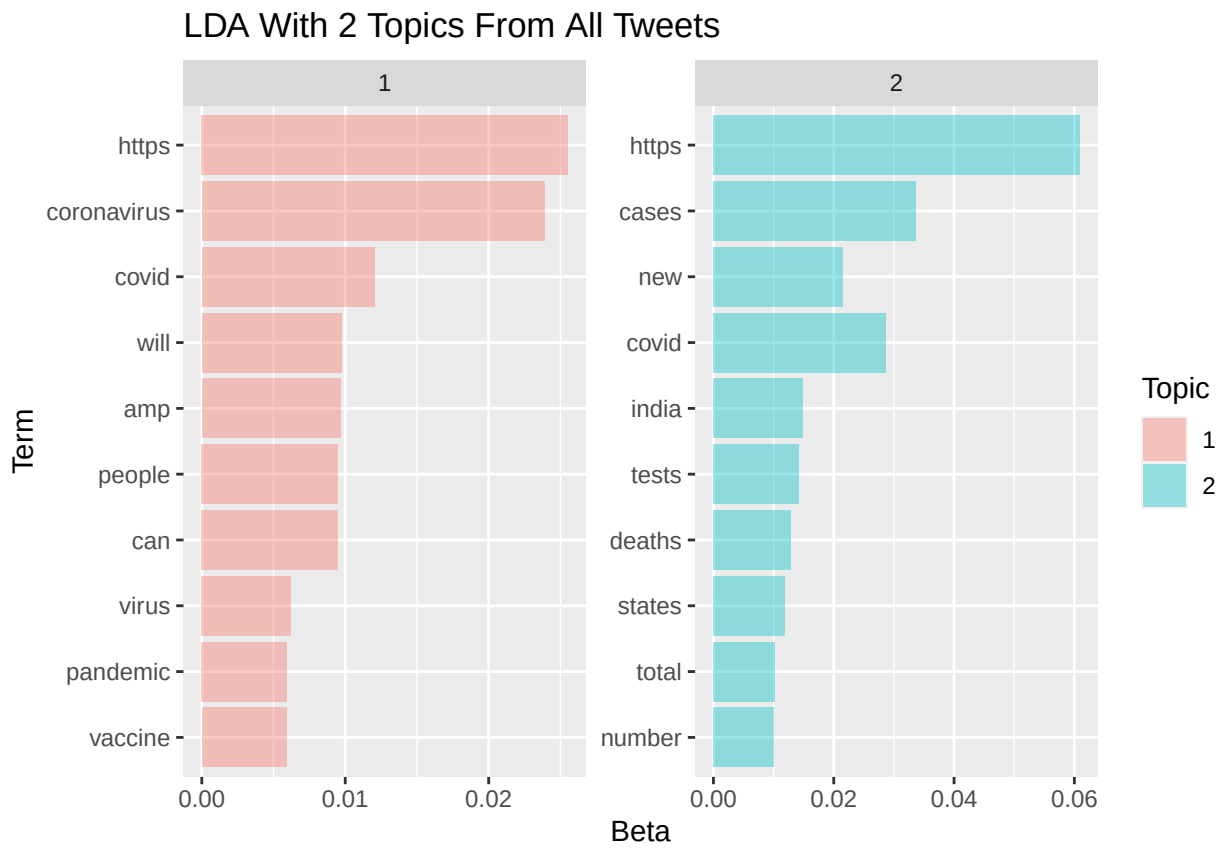
LDA is an unsupervised machine learning approach that recovers topic clusters from text. The procedure is as follows:

1. Specify the number of clusters or topics to recover, N
2. Distribute the topics via a Dirichlet Distribution.θ
3. Sample the words via a second Dirichlet Distribution, β
4. Optimize with Gibbs (Zvornicanin, 2022)

Topic Modeling Full Dataset

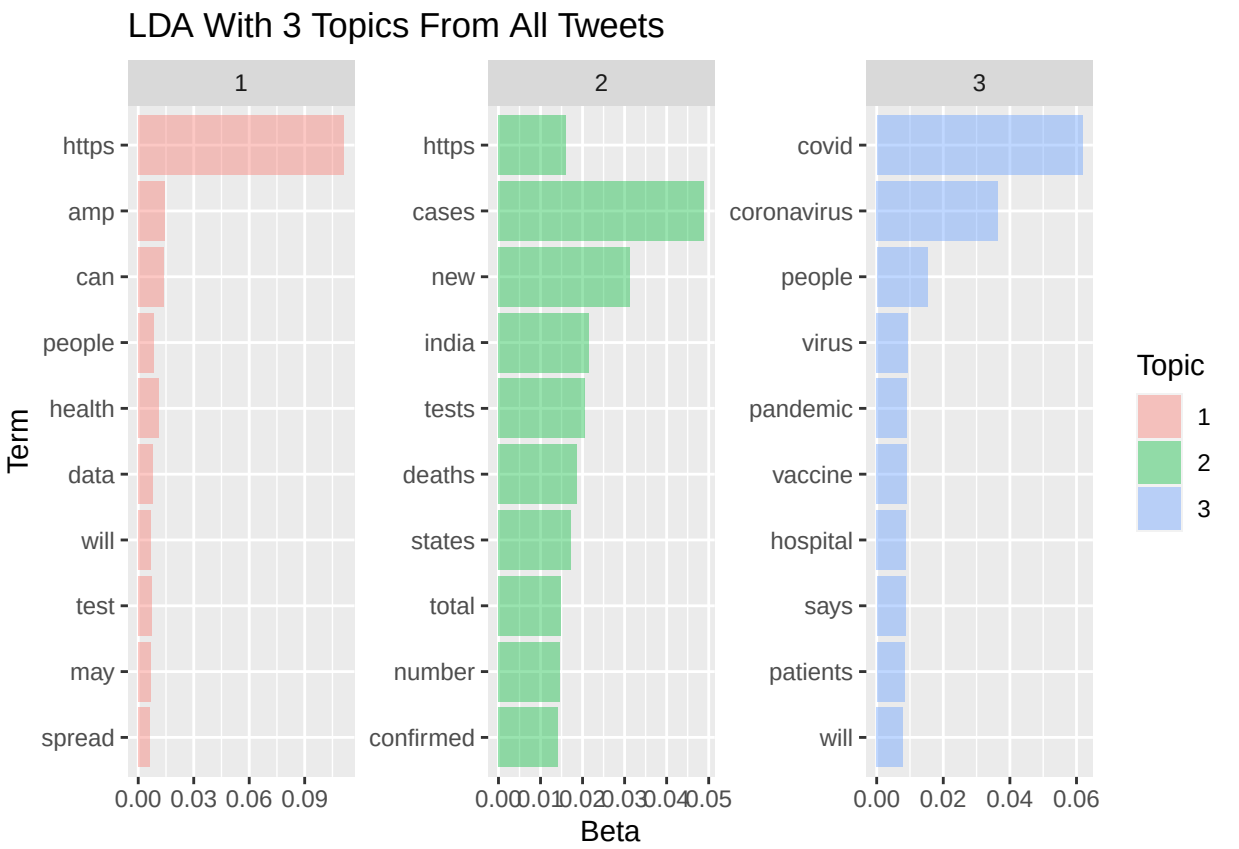
When interpreting the graphs in this section, it is critical to give attention to the Beta β. A good topic is one where the top 10 terms have a high beta while a poor topic is one where the beta is low.

```
ggplot(lda_pipeline(dtm, 2), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 2 Topics From All Tweets", fill = "Topic")
```



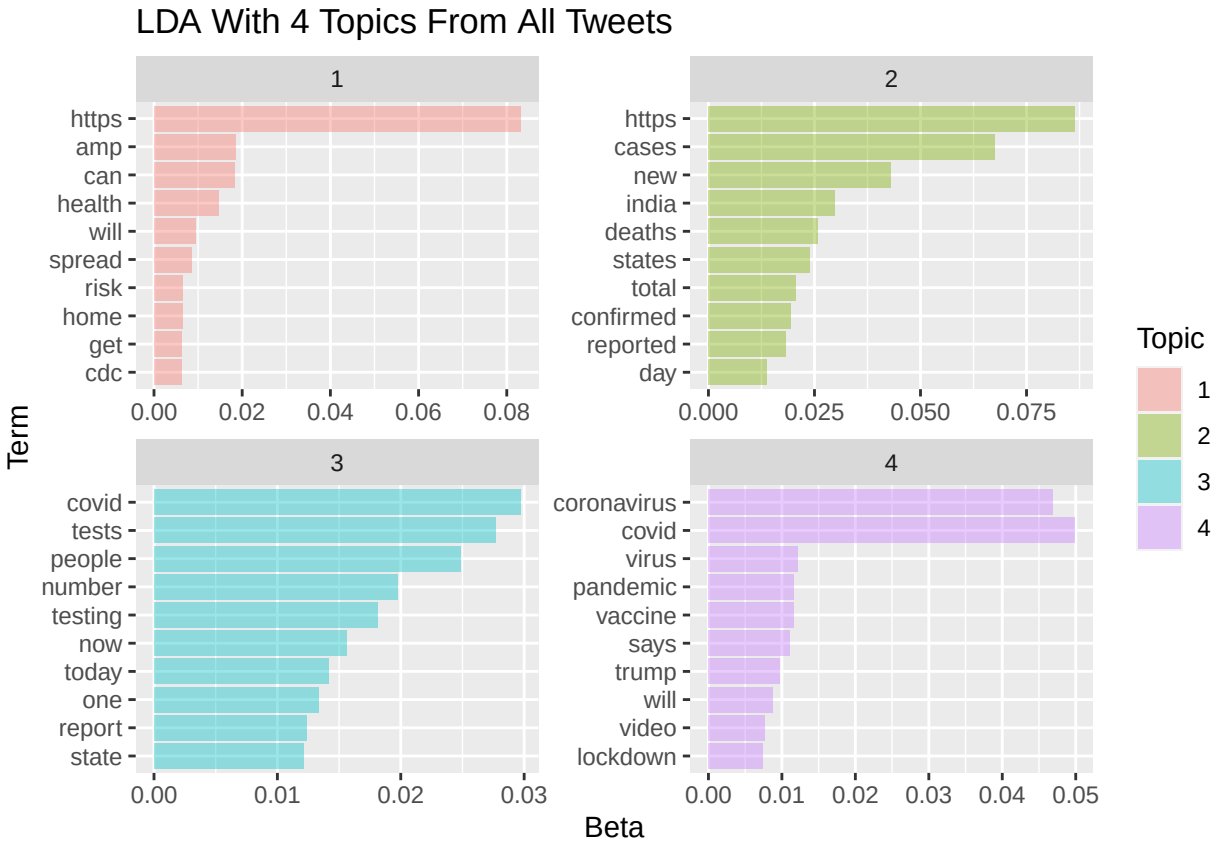
When LDA was performed with 2 clusters, a clear topic did not emerge.

```
ggplot(lda_pipeline(dtm, 3), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4) + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 3 Topics From All Tweets", fill = "Topic")
```



When the cluster K was set to 3, the topics were a little more clear. It seems topic 2 focuses on the situation in India

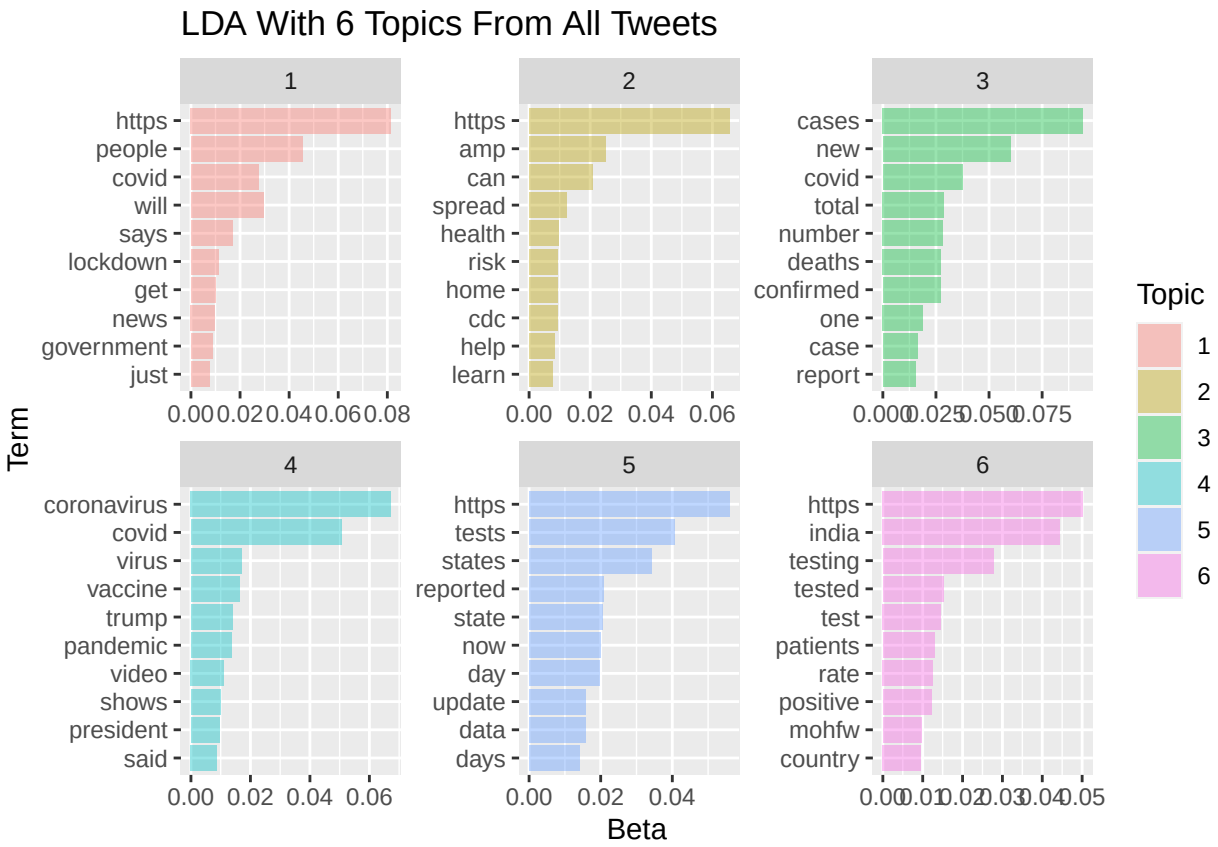
```
ggplot(lda_pipeline(dtm, 4), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 4 Topics From All Tweets", fill = "Topic")
```



When K was set to 4 the following topics emerged:

1. Not really clear. Seems to have an emphasis on India and Trump
2. Covid information from sources like AMP or the CDC.
3. Seems to be a topic around the positive covid tests
4. Seems to be about the covid deaths by state

```
ggplot(lda_pipeline(dtm, 6), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 6 Topics From All Tweets", fill = "Topic")
```



When the cluster number was set to 6 the following topics emerged:

1. Seems to be focused on India
2. Seems to be focused on covid stats put out by the government
3. Seems to be focused on covid info published by the CDC or AMP
4. Seems to be covid information by state
5. Covd deaths
6. Former Pesident Trump's covid response

Topic Modeling Fake Tweets

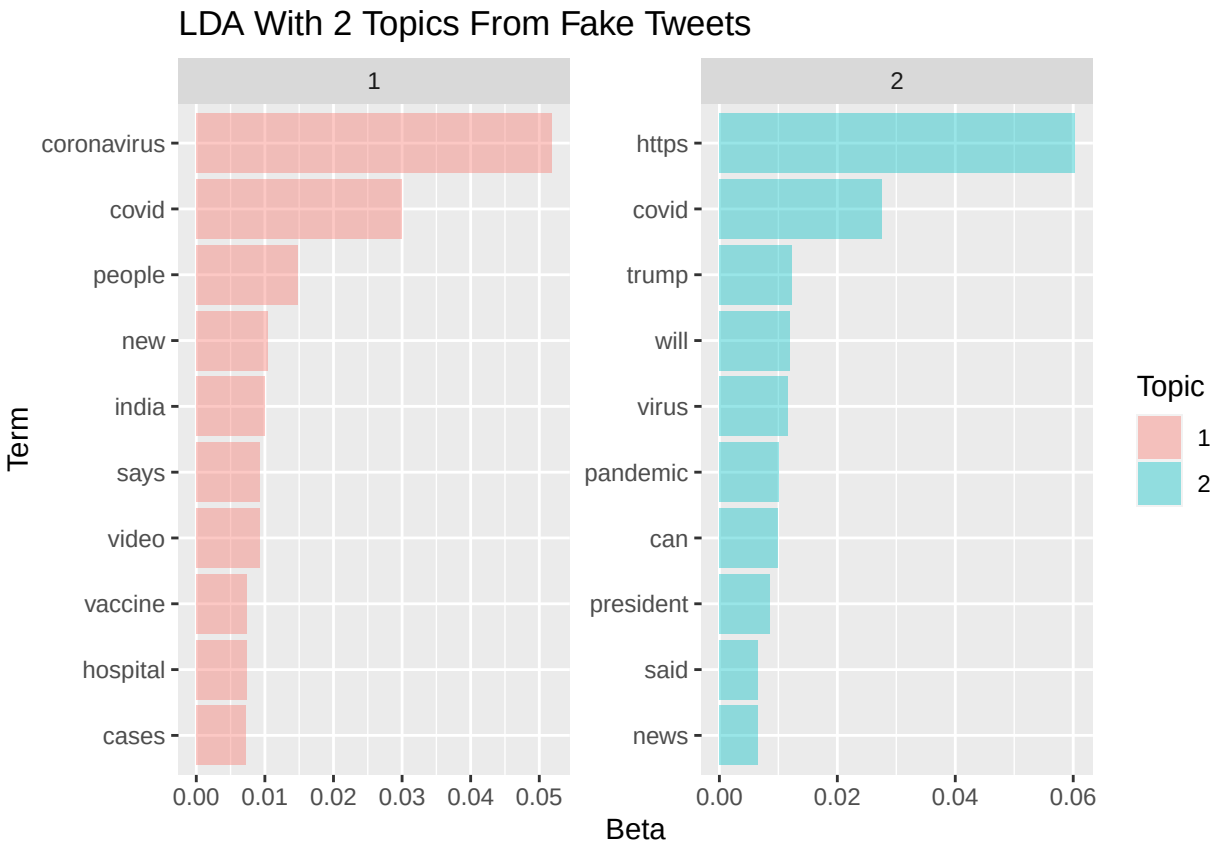
Data Loading

Subset the data to contain only the fake tweets then create a document term matrix

```
fake <- df[df$label == "fake",]
dtm <- fake %>% unnest_tokens(word, tweet) %>%count(word, id) %>% cast_dtm(id, word, n) %>% as.matrix()
```

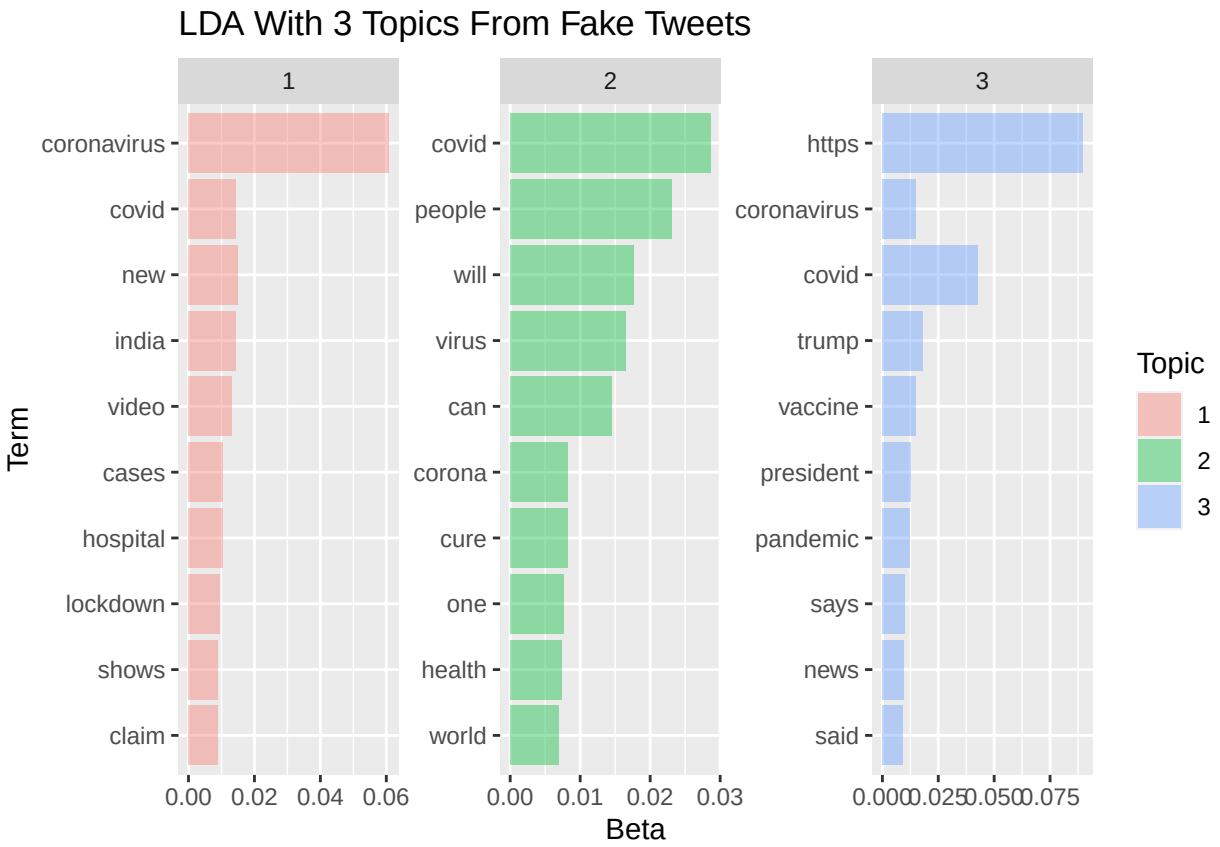
LDA Modeling

```
ggplot(lda_pipeline(dtm, 2), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 2 Topics From Fake Tweets", fill = "Topic")
```



When the number of clusters is set to 2, it seems that the first topic is focused on covid videos from the hospital while the second is covid policy

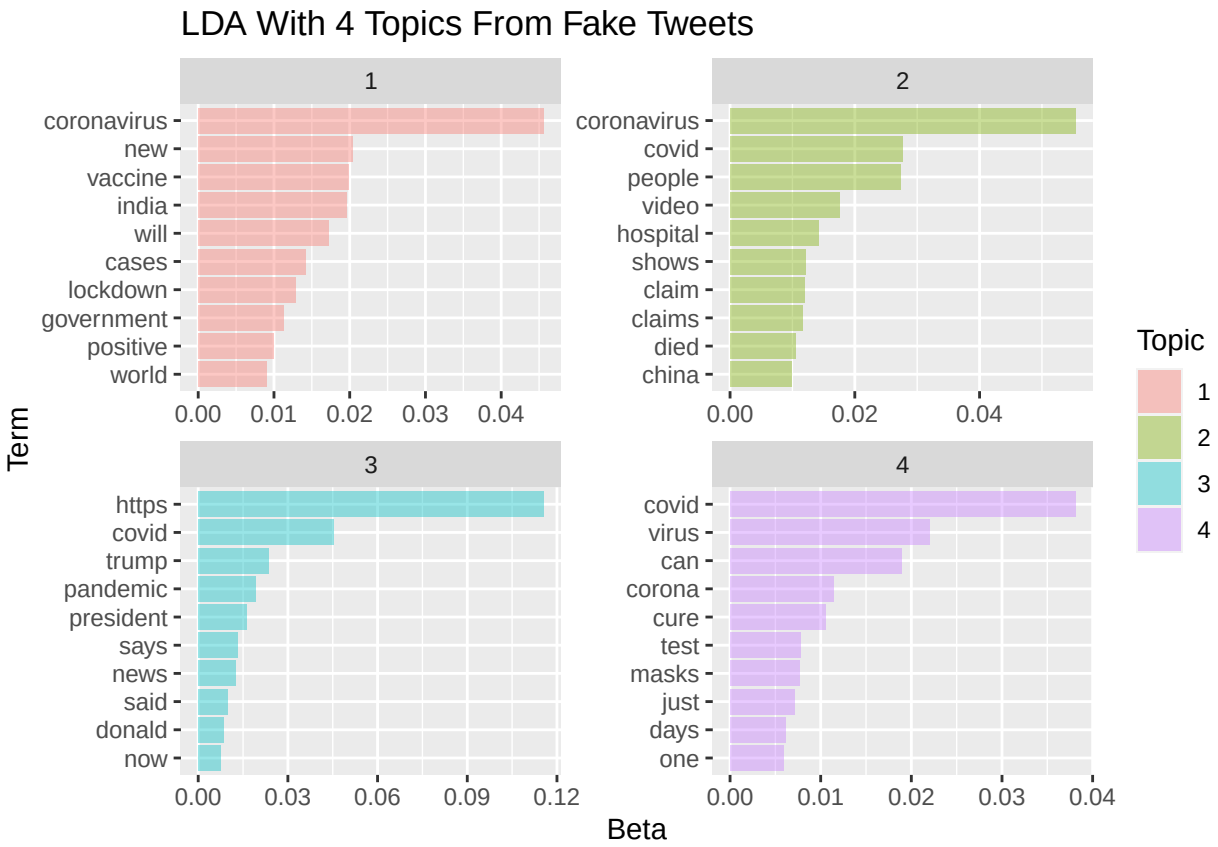
```
ggplot(lda_pipeline(dtm, 3), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4) + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 3 Topics From Fake Tweets", fill = "Topic")
```



For fake tweets and a cluster limit sent to 3, the following topics seemed to emerge:

1. Indian covid response
2. Topic 2 seems focused on the the vaccine around the world
3. Trump's response to the pandemic

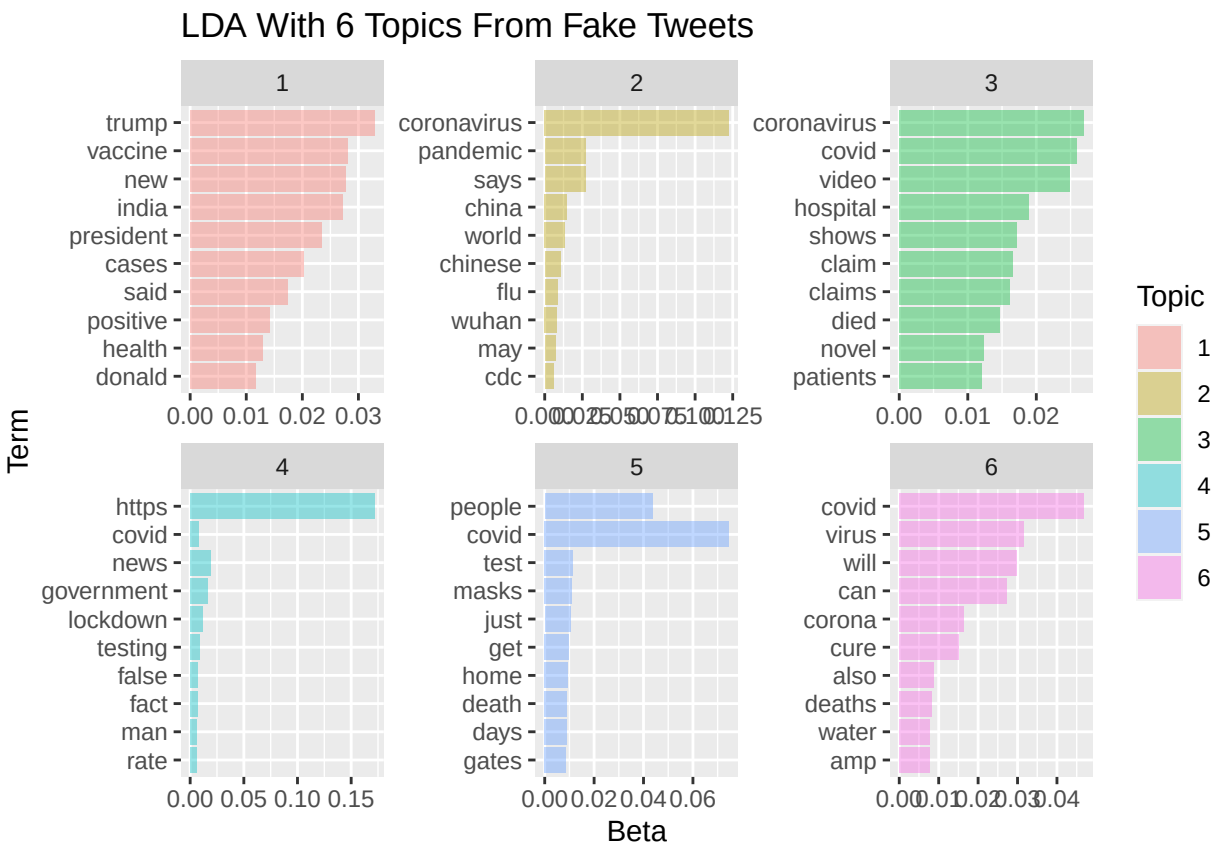
```
ggplot(lda_pipeline(dtm, 4), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 4 Topics From Fake Tweets", fill = "Topic")
```



For fake tweets and a cluster limit sent to 4, the following topics seemed to emerge:

1. President Trump's covid vaccine and attribution of the virus to China
2. Covid policy and affect on people
3. Masks and global covid numbers
4. The covid situation in India

```
ggplot(lda_pipeline(dtm, 6), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 6 Topics From Fake Tweets", fill = "Topic")
```



For fake tweets and a cluster limit sent to 6, the following topics seemed to emerge:

1. The covid mask policy
2. Covid death rate
3. President Trump's comparison of covid and the flu?
4. Covid hospital videos that circulated on facebook
5. China's covid response
6. Covid situation in India

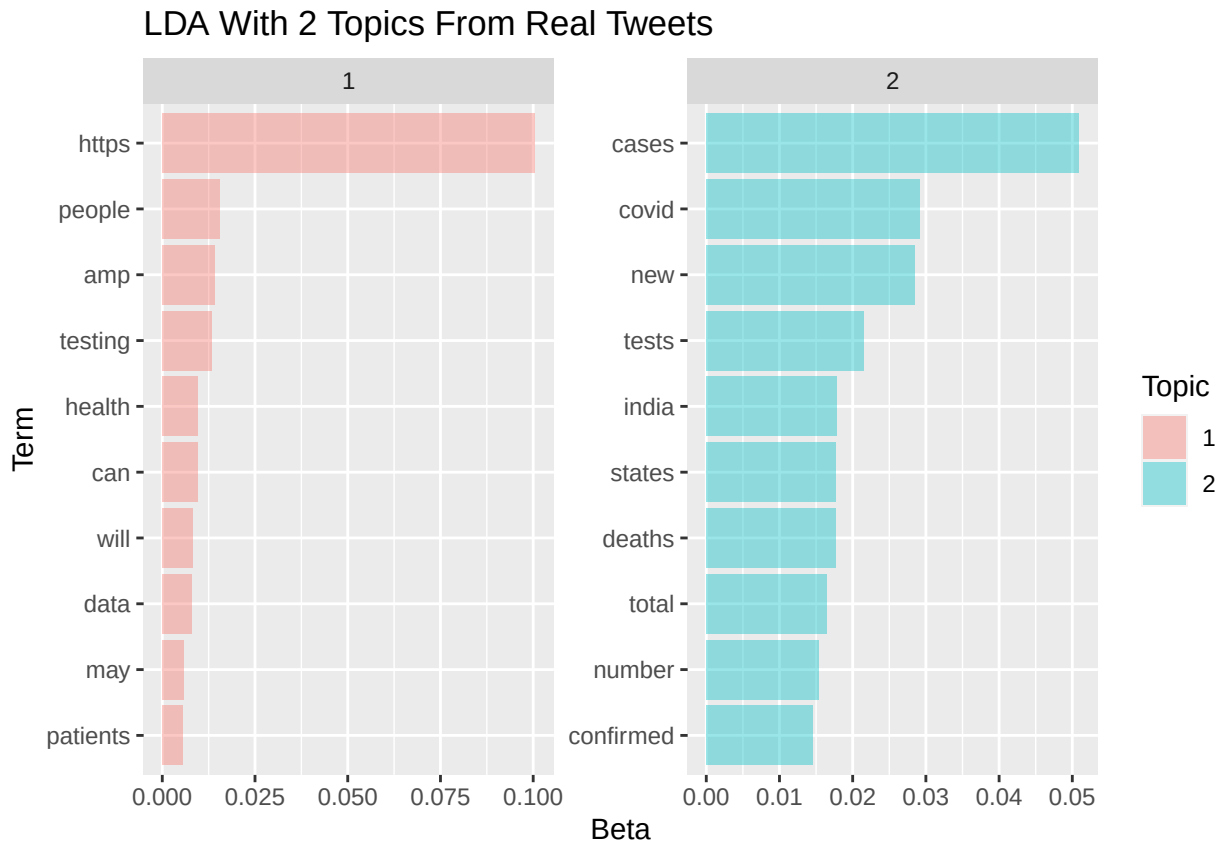
Topic Modeling Real Tweets

Subset the data to contain only the real tweets then create a document term matrix

```
real <- df[df$label == "real",]
dtm <- real %>% unnest_tokens(word, tweet) %>%count(word, id) %>% cast_dtm(id, word, n) %>% as.matrix()
```

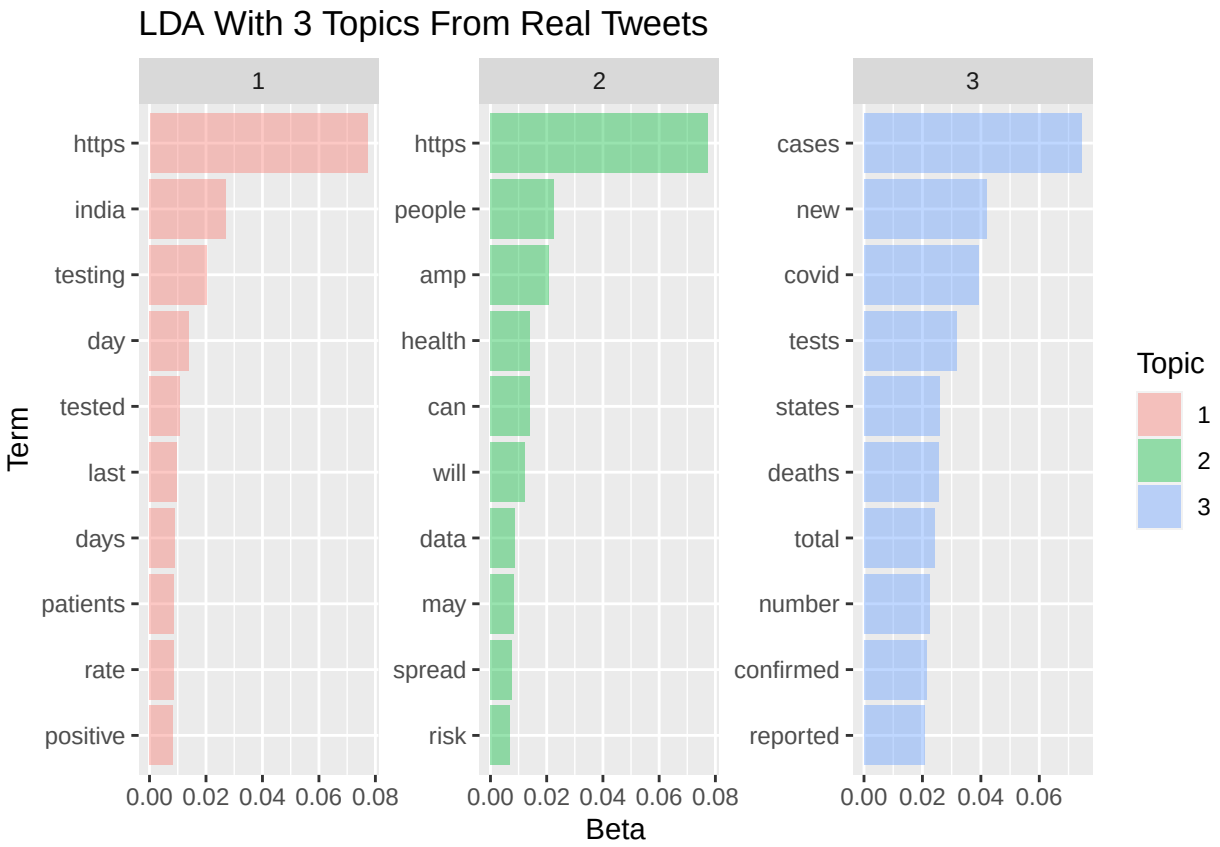
LDA Modeling

```
ggplot(lda_pipeline(dtm, 2), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 2 Topics From Real Tweets", fill = "Topic")
```



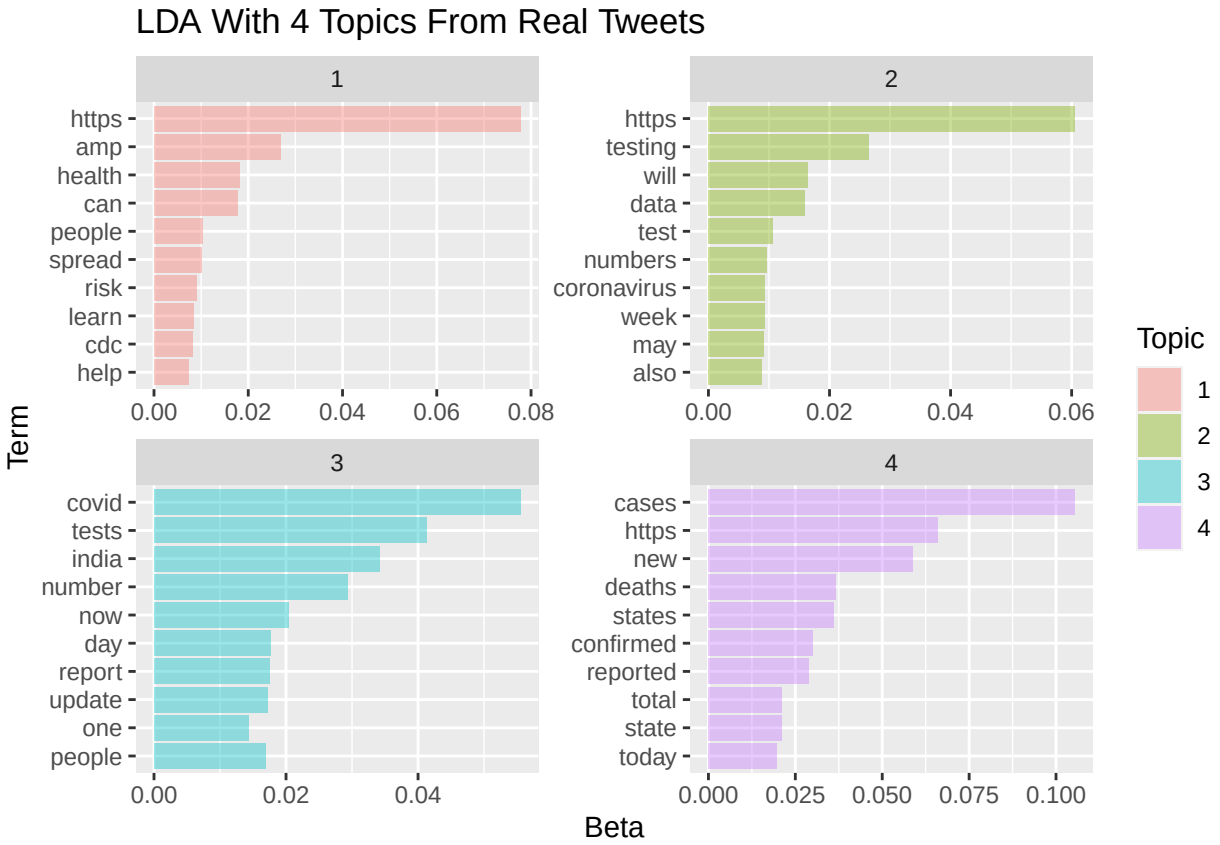
When the number of clusters is set to 2, a clear topic failed to emerge.

```
ggplot(lda_pipeline(dtm, 3), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4) + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 3 Topics From Real Tweets", fill = "Topic")
```



When the number of clusters is set to 3, topic 2 seems to have a clear focus on India, but the other two are not clear.

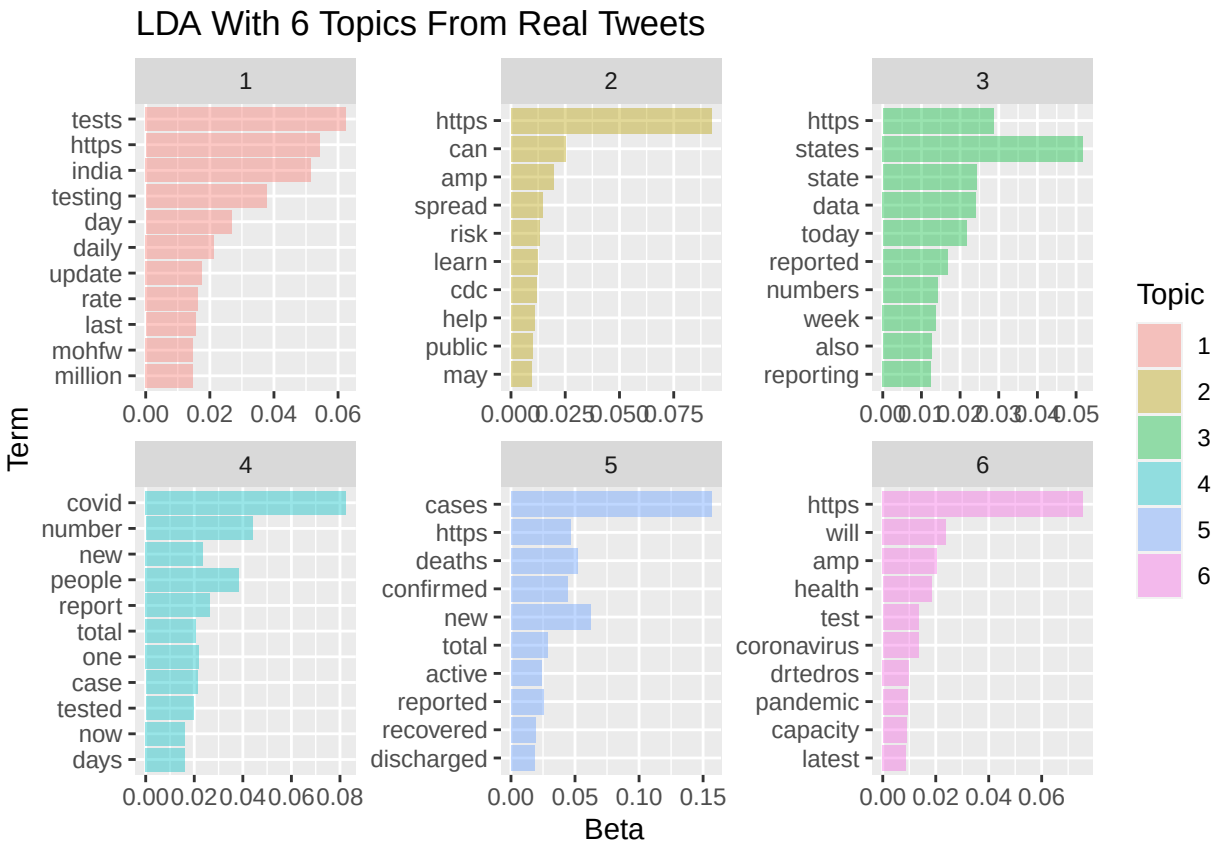
```
ggplot(lda_pipeline(dtm, 4), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 4 Topics From Real Tweets", fill = "Topic")
```



For real tweets and a cluster limit sent to 6, the following topics seemed to emerge:

1. The positive covid test data
2. The covid situation in India
3. The covid death rate
4. Covid information from sources like AMP about spread and risk

```
ggplot(lda_pipeline(dtm, 6), aes(x = ordered_term, y = beta,fill = as.factor(topic))) +
  geom_col(alpha = 0.4)  + facet_wrap(~ topic, scales = "free") + coord_flip() +
  labs(y = "Beta", x = "Term", title = "LDA With 6 Topics From Real Tweets", fill = "Topic")
```



For real tweets and a cluster limit sent to 6, the following topics seemed to emerge:

1. The covid test numbers by state
2. The current covid test numbers
3. The currenct covid situation in India
4. Covid information from sources like the CDC or AMP
5. Covid death rate?
6. No clue. . .

References

Zvornicanin, E. (2022). *Topic Modeling and Latent Dirichlet Allocation.* Retrieved From: https://datascienceplus.com/topic-modeling-and-latent-dirichlet-allocation-lda/