Latent Dirichlet allocation

INTRODUCTION TO TEXT ANALYSIS IN R

Maham Faisal Khan

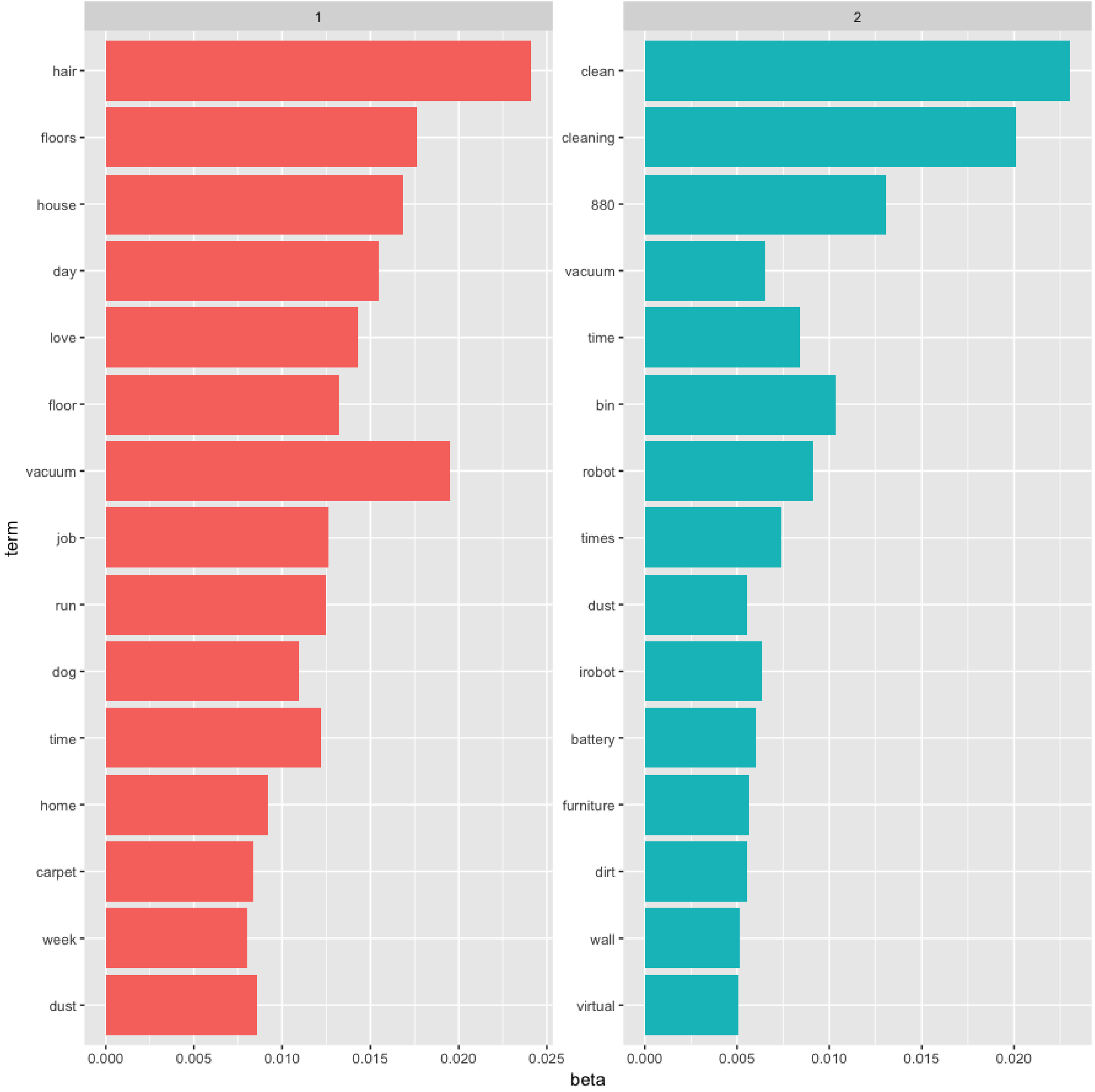
Senior Data Science Content Developer

Unsupervised learning

Some more natural language processing (NLP) vocabulary:

- Latent Dirichlet allocation (LDA) is a standard topic model
- A collection of documents is known as a corpus
- Bag-of-words is treating every word in a document separately
- Topic models find patterns of words appearing together
- Searching for patterns rather than predicting is known as unsupervised learning

Word probabilities



Clustering vs. topic modeling

Clustering

- Clusters are uncovered based on distance, which is continuous.
- Every object is assigned to a single cluster.

Topic Modeling

- Topics are uncovered based on word frequency, which is discrete.
- Every document is a mixture (i.e., partial member) of every topic.

Let's practice!

INTRODUCTION TO TEXT ANALYSIS IN R

Document term matrices

INTRODUCTION TO TEXT ANALYSIS IN R

Maham Faisal Khan
Senior Data Science Content Developer

Matrices and sparsity

```
sparse_review
```

```
    Terms
Docs admit ago albeit amazing angle awesome
   4     1   0      1       0     0       0
   5     0   1      0       1     1       0
   3     0   0      0       0     0       1
   2     0   0      0       0     0       0
```

Using cast_dtm()

```r
tidy_review %>%
  count(word, id) %>%
  cast_dtm(id, word, n)
```

```
<<DocumentTermMatrix (documents: 1791, terms: 9669)>>
Non-/sparse entries: 62766/17252622
Sparsity           : 100%
Maximal term length: NA
Weighting          : term frequency (tf)
```

Using as.matrix()

```
dtm_review <- tidy_review %>%
  count(word, id) %>%
  cast_dtm(id, word, n) %>%
  as.matrix()
dtm_review[1:4, 2000:2004]
```

```
      Terms
Docs   consecutive consensus consequences considerable considerably
  223            0         0            0            0            0
  615            0         0            0            0            0
  1069           0         0            0            0            0
  425            0         0            0            0            0
```

Let's practice!

INTRODUCTION TO TEXT ANALYSIS IN R

Running topic models

INTRODUCTION TO TEXT ANALYSIS IN R

Maham Faisal Khan

Senior Data Science Content Developer

Using LDA()

```r
library(topicmodels)
lda_out <- LDA(
  dtm_review,
  k = 2,
  method = "Gibbs",
  control = list(seed = 42)
)
```

LDA() output

```
lda_out
```

```
A LDA_Gibbs topic model with 2 topics.
```

Using glimpse()

```
glimpse(lda_out)
```

```
Formal class 'LDA_Gibbs' [package "topicmodels"] with 16 slots
  ..@ seedwords      : NULL
  ..@ z              : int [1:75670] 1 2 2 1 1 2 1 1 2 2 ...
  ..@ alpha          : num 25
  ..@ call           : language LDA(x = dtm_review, k = 2, method = "Gibbs", ...
  ..@ Dim            : int [1:2] 1791 9668
  ..@ control        :Formal class 'LDA_Gibbscontrol' [package "topicmodels"] ...
  ..@ beta           : num [1:2, 1:17964] -8.81 -10.14 -9.09 -8.43 -12.53 ...
  ...
```

Using tidy()

```
lda_topics <- lda_out %>%
  tidy(matrix = "beta")
lda_topics %>%
  arrange(desc(beta))
```

```
# A tibble: 19,336 x 3
   topic term        beta
   <int> <chr>      <dbl>
 1     1 hair      0.0241
 2     2 clean     0.0231
 3     2 cleaning  0.0201
# … with 19,333 more rows
```

Let's practice!

INTRODUCTION TO TEXT ANALYSIS IN R

Interpreting topics

INTRODUCTION TO TEXT ANALYSIS IN R

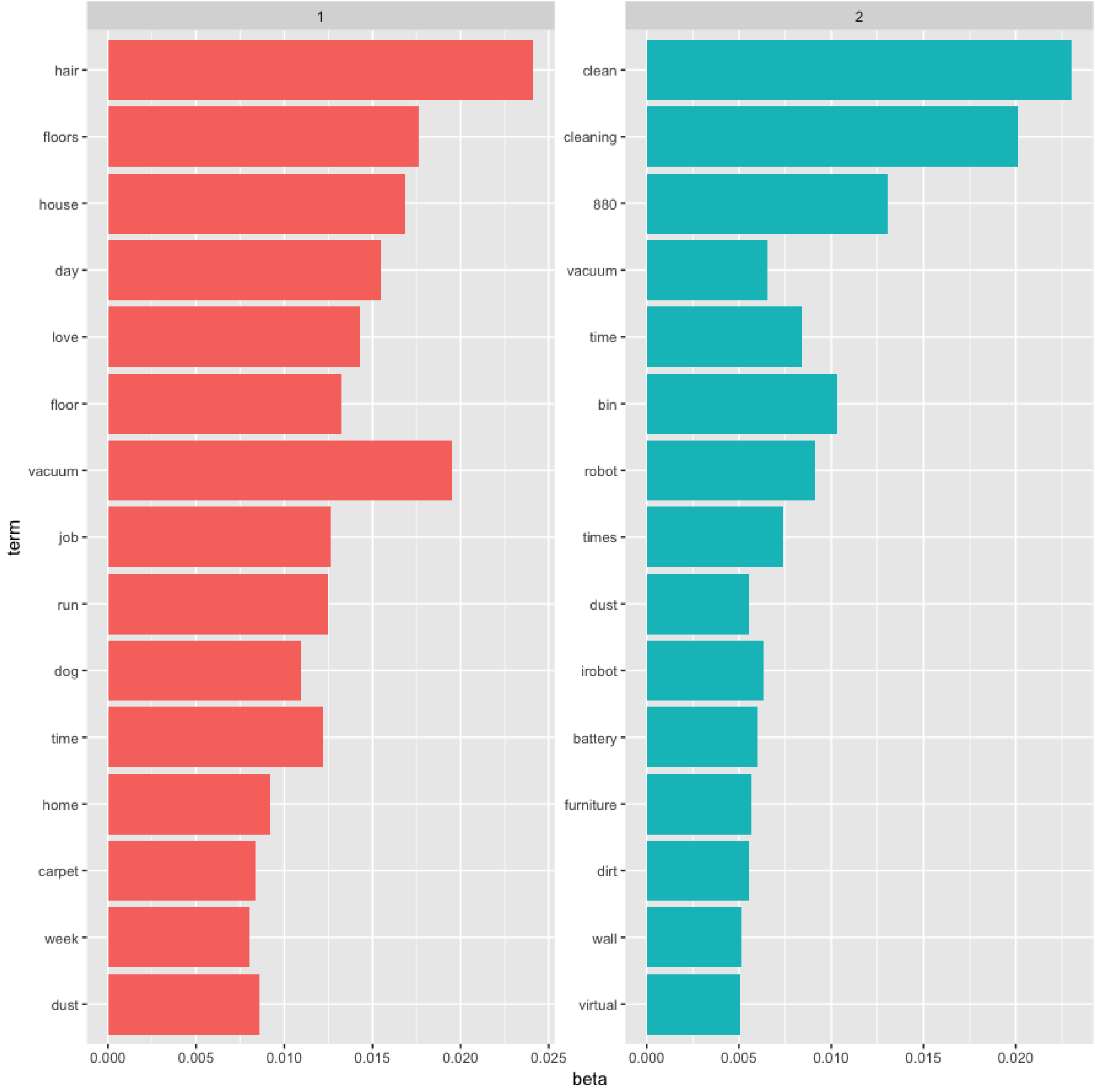
Maham Faisal Khan
Senior Data Science Content Developer

Two topics

```
lda_topics <- LDA(
  dtm_review,
  k = 2,
  method = "Gibbs",
  control = list(seed = 42)
) %>%
  tidy(matrix = "beta")
word_probs <- lda_topics %>%
  group_by(topic) %>%
  slice_max(beta, n = 15) %>%
  ungroup() %>%
  mutate(term2 = fct_reorder(term, beta))
```

Two topics

```
ggplot(
  word_probs,
  aes(
    term2,
    beta,
    fill = as.factor(topic)
  )
) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip()
```

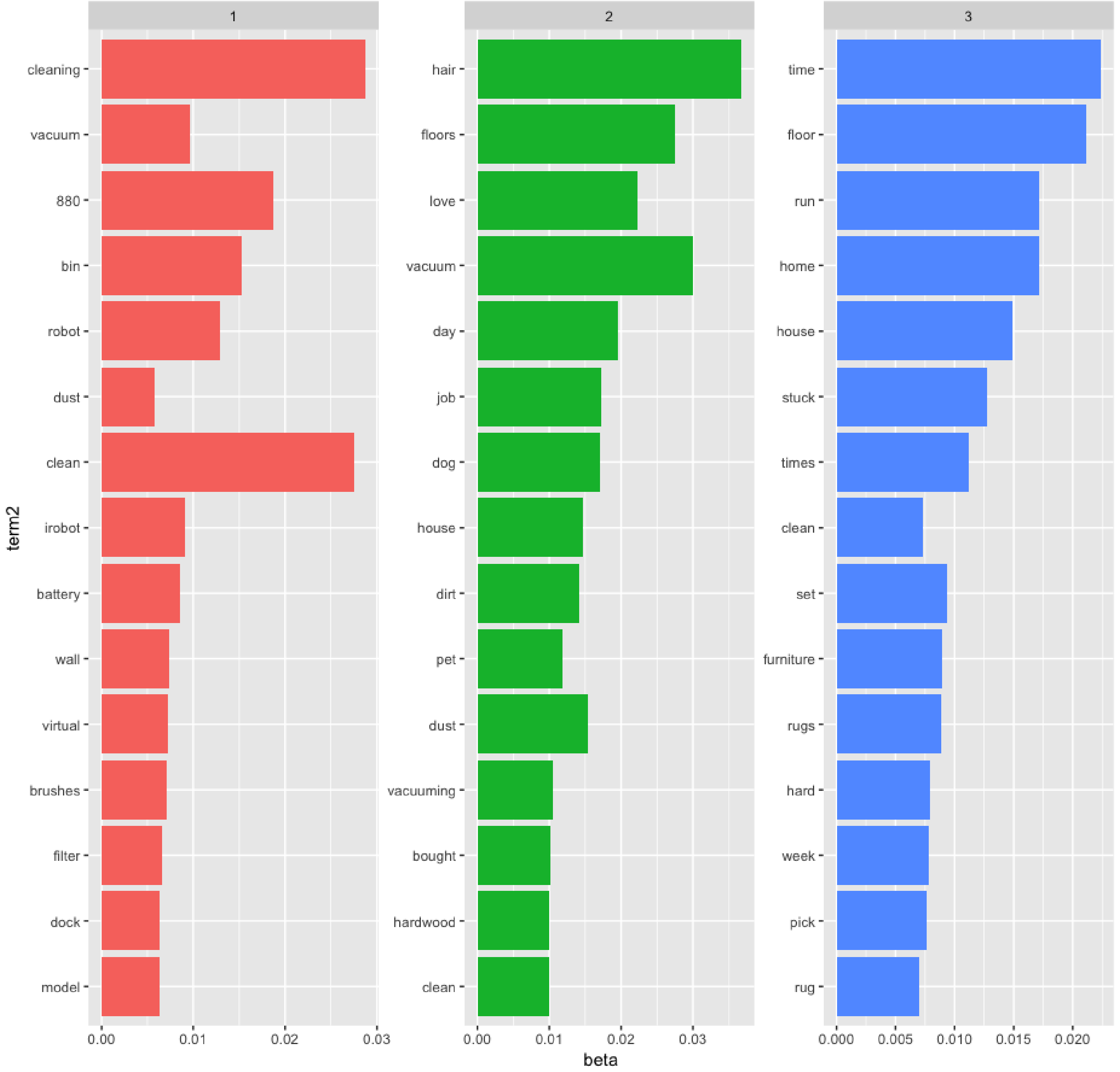


Three topics

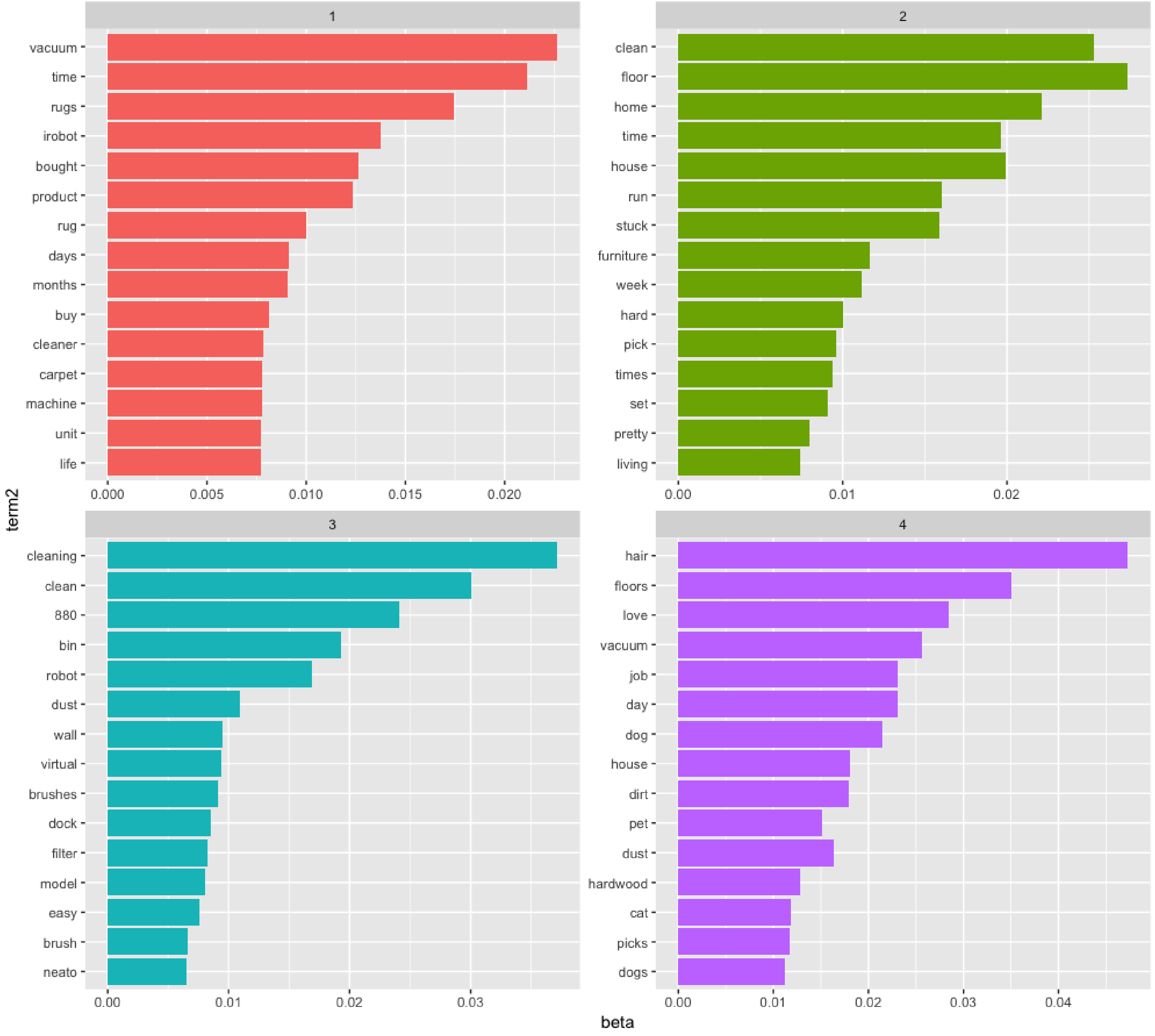
```r
lda_topics2 <- LDA(
  dtm_review,
  k = 3,
  method = "Gibbs",
  control = list(seed = 42)
) %>%
  tidy(matrix = "beta")
word_probs2 <- lda_topics2 %>%
  group_by(topic) %>%
  slice_max(beta, n = 15) %>%
  ungroup() %>%
  mutate(term2 = fct_reorder(term, beta))
```

Three topics

```
ggplot(
  word_probs2,
  aes(
    term2,
    beta,
    fill = as.factor(topic)
  )
) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip()
```



Four topics


1
vacuum
time
rugs
irobot
bought
product
rug
days
months
buy
cleaner
carpet
machine
unit
life
0.000
0.005
0.010
0.015
0.020
2
clean
floor
home
time
house
run
stuck
furniture
week
hard
pick
times
set
pretty
living
0.00
0.01
0.02
3
cleaning
clean
880
bin
robot
dust
wall
virtual
brushes
dock
filter
model
easy
brush
neato
0.00
0.01
0.02
0.03
4
hair
floors
love
vacuum
job
day
dog
house
dirt
pet
dust
hardwood
cat
picks
dogs
0.00
0.01
0.02
0.03
0.04
term2
beta

The art of model selection

- Adding topics that are different is good
- If we start repeating topics, we've gone too far
- Name the topics based on the combination of high-probability words

Let's practice!

INTRODUCTION TO TEXT ANALYSIS IN R

Wrap-up

INTRODUCTION TO TEXT ANALYSIS IN R

Maham Faisal Khan

Senior Data Science Content Developer

Summary

- Tokenizing text and removing stop words
- Visualizing word counts
- Conducting sentiment analysis
- Running and interpreting topic models

Next steps

Other DataCamp courses:

- Sentiment Analysis in R: The Tidy Way
- Topic Modeling in R

Additional resource:

- Text Mining with R

All the best!

INTRODUCTION TO TEXT ANALYSIS IN R