PM 566: Lab 06

AUTHOR
Tarun Mahesh

Step 1: Package Setup

Load in dplyr, ggplot2 and tidytext.

```
suppressPackageStartupMessages(library(dplyr))
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(tidytext))
suppressPackageStartupMessages(library(tidyverse))
```

Step 2: Read in Medical Transcriptions

Load in the cleaned data from the USCbiostats/data-science-data repo:

```
library(readr)
library(dplyr)
mt_samples <- read_csv("https://raw.githubusercontent.com/USCbiostats/data-science-data/master/00
```

```
New names:
• `` -> `...1`
```

```
mt_samples <- mt_samples |>
  select(description, medical_specialty, transcription)
head(mt_samples)
```

```
# A tibble: 6 × 3
  description                                     medical_specialty transcription
  <chr>                                           <chr>             <chr>
1 A 23-year-old white female presents with comp… Allergy / Immuno… "SUBJECTIVE:…
2 Consult for laparoscopic gastric bypass.        Bariatrics        "PAST MEDICA…
3 Consult for laparoscopic gastric bypass.        Bariatrics        "HISTORY OF …
4 2-D M-Mode. Doppler.                            Cardiovascular /… "2-D M-MODE:…
```

```
5 2-D Echocardiogram                              Cardiovascular /… "1.  The lef…
6 Morbid obesity.  Laparoscopic antecolic anteg… Bariatrics        "PREOPERATIV…
```

QUESTION 1:

What specialities do we have?

Use the `count()` function from `dplyr` to figure out how many different categories we have in the data. Are these categories related? Overlapping? Evenly distributed?

```
mt_samples %>%
  count(medical_specialty, sort = TRUE)
```

```
# A tibble: 40 × 2
   medical_specialty                 n
   <chr>                         <int>
 1 Surgery                        1103
 2 Consult - History and Phy.      516
 3 Cardiovascular / Pulmonary      372
 4 Orthopedic                      355
 5 Radiology                       273
 6 General Medicine                259
 7 Gastroenterology                230
 8 Neurology                       223
 9 SOAP / Chart / Progress Notes   166
10 Obstetrics / Gynecology         160
# ℹ 30 more rows
```

There are 40 categories, and are unique, however they are interrelated. The observations are not evenly distributed across the categories.

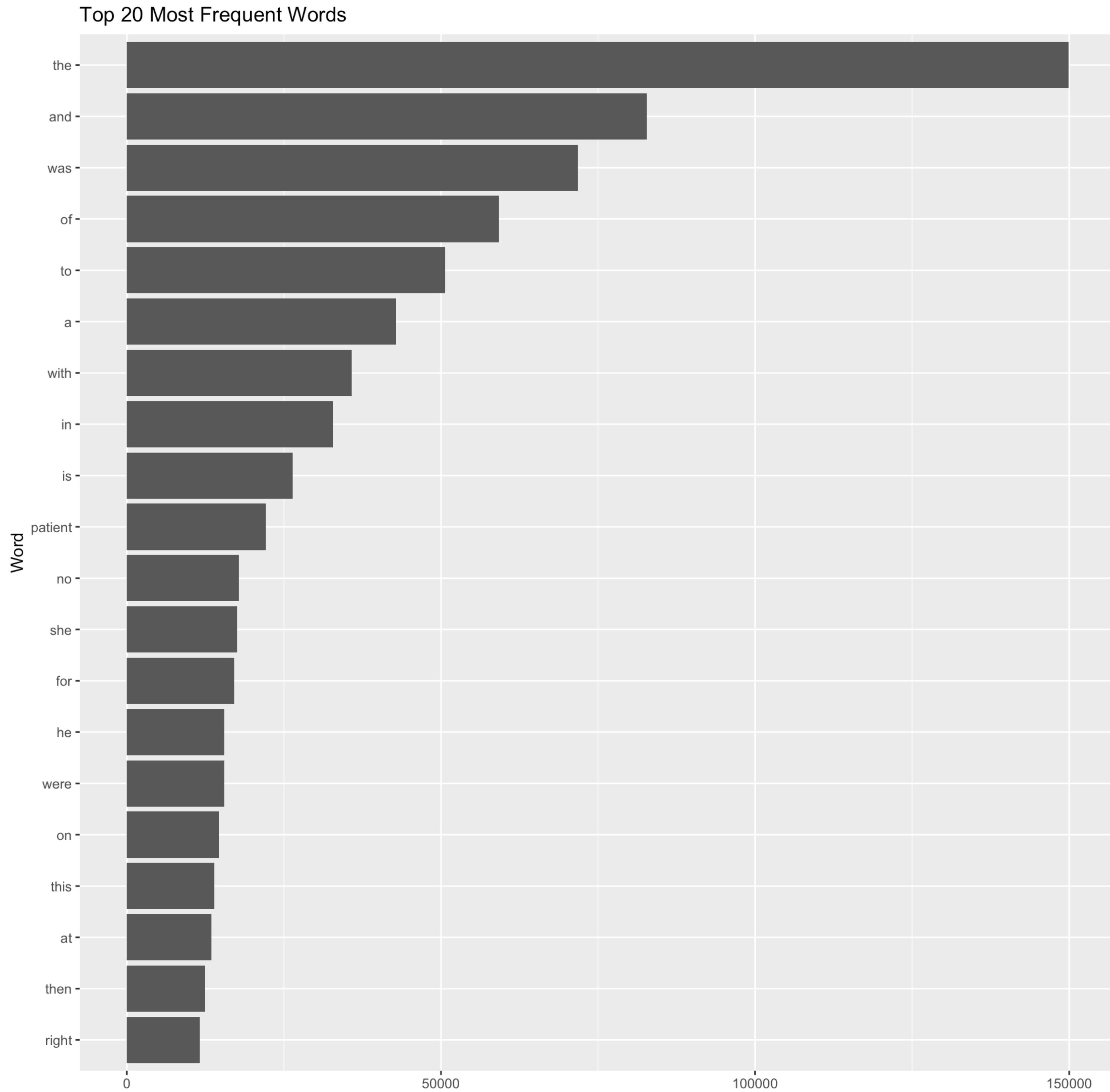
QUESTION 2:

- Tokenize the the words in the `transcription` column

- Count the number of times each token appears
- Visualize the top 20 most frequent words

Explain what we see from this result. Does it makes sense? What insights (if any) do we get?

```
mt_samples %>%
  unnest_tokens(word, transcription) %>%
  count(word, sort = TRUE) %>%
  top_n(20, n) %>%
  ggplot(aes(x = reorder(word, n), y = n)) +
  geom_col() +
  coord_flip() +
  labs(x = "Word", y = "Count", title = "Top 20 Most Frequent Words")
```


Top 20 Most Frequent Words
Word
the
and
was
of
to
a
with
in
is
patient
no
she
for
he
were
on
this
at
then
right
0
50000
100000
150000

Count

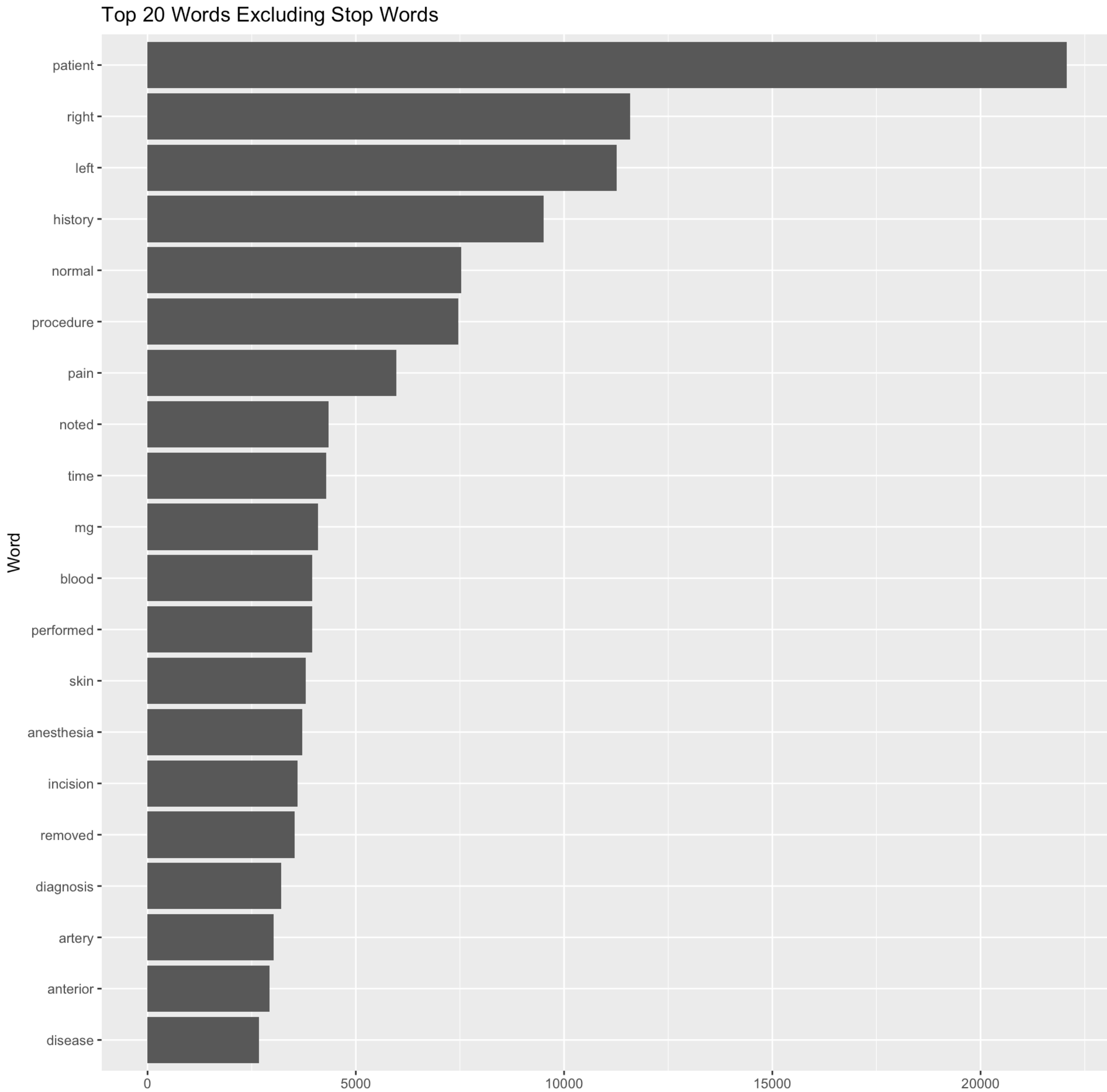
The most common words are stop words like “the”, “a”, “an”, “was” etc., which is expected and sensible but does not provide us any real insight into the data.

QUESTION 3:

- Re-do the visualization but remove stop words before making it
- Bonus points if you remove numbers as well

What do we see know that we have removed stop words? Does it give us a better idea of what the text is about?

```
mt_samples %>%
  unnest_tokens(word, transcription) %>%
  anti_join(stop_words %>% filter(!word %in% c("right", "left")), by = "word") %>%
  filter(!grepl("^[0-9]+$", word)) %>%
  count(word, sort = TRUE) %>%
  top_n(20, n) %>%
  ggplot(aes(x = reorder(word, n), y = n)) +
  geom_col() +
  coord_flip() +
  labs(x = "Word", y = "Count", title = "Top 20 Words Excluding Stop Words")
```


Top 20 Words Excluding Stop Words
patient
right
left
history
normal
procedure
pain
noted
time
mg
blood
performed
skin
anesthesia
incision
removed
diagnosis
artery
anterior
disease
Word
0
5000
10000
15000
20000

Count

After removing the stop words, the most common words are more procedure/patient centric. The words being used seem to signify that the data is largely descriptive of patient vitals (time, mg, blood, normal, disease etc.) or surgery metadata.

QUESTION 4:

Repeat question 2, but this time tokenize into bi-grams. How does the result change if you look at tri-grams?

```
#Bigrams
mt_samples %>%
  unnest_tokens(bigram, transcription, token = "ngrams", n = 2) %>%
  count(bigram, sort = TRUE) %>%
  top_n(20, n)
```

```
# A tibble: 20 × 2
   bigram          n
   <chr>       <int>
 1 the patient 20307
 2 of the      19062
 3 in the      12790
 4 to the      12374
 5 was then     6956
 6 and the      6350
 7 patient was  6293
 8 the right    5509
 9 on the       5241
10 the left     4860
11 with a       4857
12 history of   4537
13 to be        4345
14 is a         4014
15 with the     4002
16 there is     3950
17 at the       3657
```

```
18 there was    3334
19 patient is   3332
20 was placed   3328
```

```
#Trigrams
mt_samples %>%
  unnest_tokens(trigram, transcription, token = "ngrams", n = 3) %>%
  count(trigram, sort = TRUE) %>%
  top_n(20, n)
```

```
# A tibble: 22 × 2
   trigram                n
   <chr>              <int>
 1 the patient was     6104
 2 the patient is      3075
 3 as well as          2243
 4 there is no         1678
 5 the operating room  1532
 6 patient is a        1491
 7 prepped and draped  1490
 8 was used to         1480
 9 and draped in       1372
10 at this time        1333
# ℹ 12 more rows
```

The trigrams are largely more descriptive than the bigrams because the bigrams still contain a lot of stop words that do not provide data insights. The trigrams capture descriptive phrases used when documenting routine medical procedures.

QUESTION 5:

Using the results you got from Question 4, pick a word and count the words that appear before and after it.

```
mt_samples %>%
  unnest_tokens(bigram, transcription, token = "ngrams", n = 2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
```

```
  filter(word1 == "operating" | word2 == "operating") %>%
  count(word1, word2, sort = TRUE)
```

```
# A tibble: 46 × 3
   word1     word2          n
   <chr>     <chr>      <int>
 1 the       operating   2000
 2 operating room        1594
 3 operating table        310
 4 operating microscope   107
 5 operating suite         78
 6 to        operating     47
 7 operating field         15
 8 on        operating     12
 9 inpatient operating     11
10 operating theater       11
# ℹ 36 more rows
```

QUESTION 6:

Which words are most used in each of the specialties? You can use `group_by()` and `top_n()` from `dplyr` to have the calculations be done within each specialty. Remember to remove stop words. What are the 5 most-used words for each specialty?

```
mt_samples %>%
  unnest_tokens(word, transcription) %>%
  anti_join(stop_words %>% filter(!word %in% c("right", "left")), by = "word") %>%
  filter(!grepl("^[0-9]+$", word)) %>%
  group_by(medical_specialty) %>%
  count(word, sort = TRUE) %>%
  top_n(5, n) %>%
  arrange(medical_specialty, desc(n))
```

```
# A tibble: 208 × 3
# Groups:   medical_specialty [40]
   medical_specialty    word          n
```

```
     <chr>                 <chr>     <int>
 1 Allergy / Immunology history      38
 2 Allergy / Immunology noted        23
 3 Allergy / Immunology patient      22
 4 Allergy / Immunology allergies    21
 5 Allergy / Immunology nasal        13
 6 Allergy / Immunology past         13
 7 Autopsy              right       108
 8 Autopsy              left         83
 9 Autopsy              inch         59
10 Autopsy              neck         55
# ℹ 198 more rows
```

QUESTION 7:

Find your own insight in the data:

Ideas:

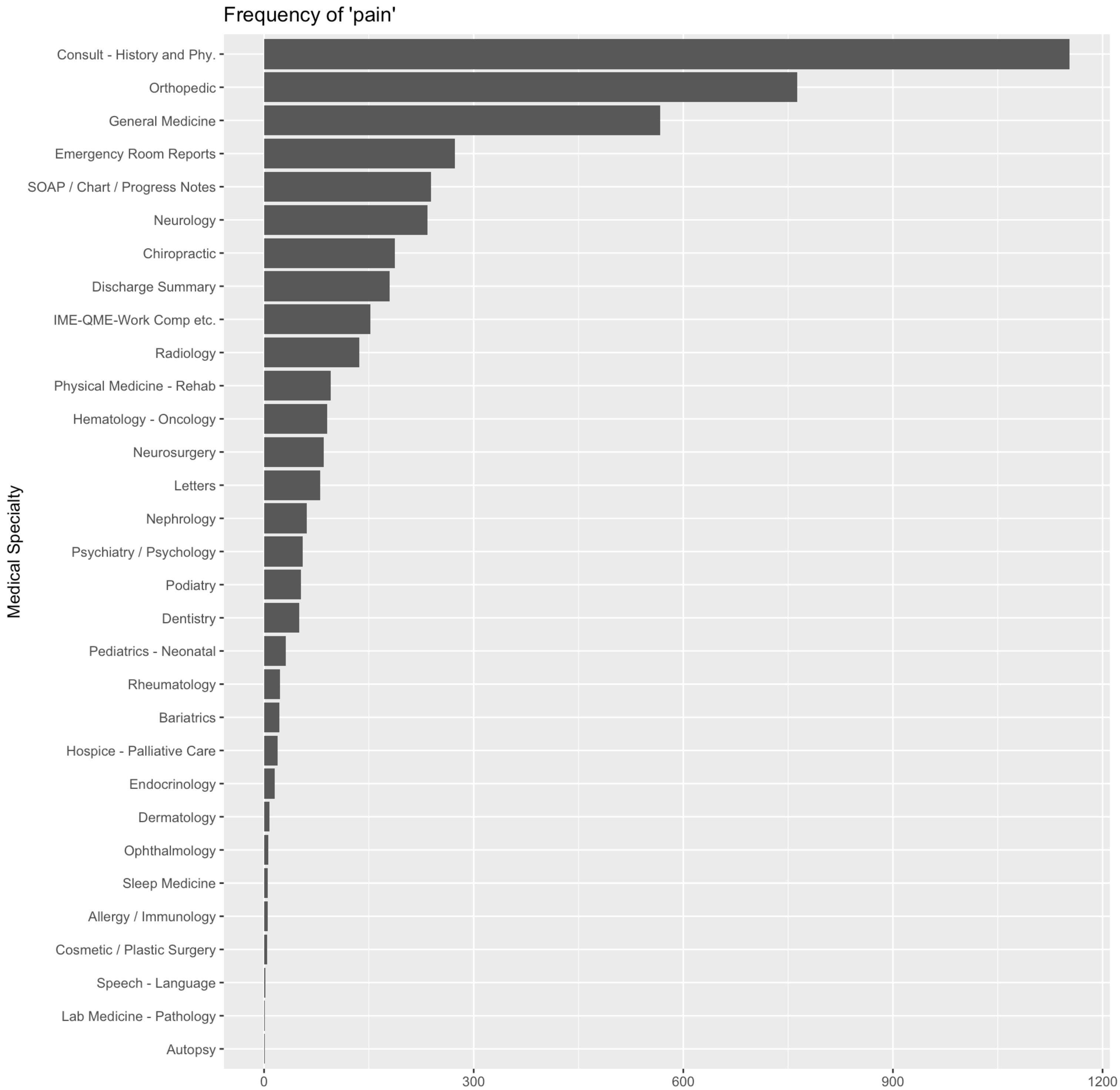
- Interesting n-grams
- See if certain words are used more in some specialties than others

My insight: Visual of use of word “pain” across different specialities

```r
pain_used <- mt_samples %>%
  unnest_tokens(word, transcription) %>%
  anti_join(stop_words %>% filter(!word %in% c("right", "left")), by = "word") %>%
  filter(!grepl("^[0-9]+$", word)) %>%
  group_by(medical_specialty) %>%
  summarize(pain_count   = sum(word == "pain")) %>%
  arrange(desc(pain_count))

pain_used %>%
  filter(pain_count > 0) %>%
  ggplot(aes(x = reorder(medical_specialty, pain_count), y = pain_count)) +
  geom_col() +
```

```
  coord_flip() +
  labs(x = "Medical Specialty", y = "Mentions of 'pain'", title = "Frequency of 'pain'")
```



Frequency of 'pain'
Medical Specialty
Consult - History and Phy.
Orthopedic
General Medicine
Emergency Room Reports
SOAP / Chart / Progress Notes
Neurology
Chiropractic
Discharge Summary
IME-QME-Work Comp etc.
Radiology
Physical Medicine - Rehab
Hematology - Oncology
Neurosurgery
Letters
Nephrology
Psychiatry / Psychology
Podiatry
Dentistry
Pediatrics - Neonatal
Rheumatology
Bariatrics
Hospice - Palliative Care
Endocrinology
Dermatology
Ophthalmology
Sleep Medicine
Allergy / Immunology
Cosmetic / Plastic Surgery
Speech - Language
Lab Medicine - Pathology
Autopsy
0
300
600
900
1200

Mentions of 'pain'