Assignment 3

Sungjoo Cho, Catherine (Kate) Lamoreaux

2023-10-22

Packages

```
library(xml2)
library(rvest)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()         masks stats::filter()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()            masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(robotstxt)
library(janeaustenr)
library(dplyr)
library(stringr)
library(tidytext)
library(dplyr)
library(tidytext)
```

Web Scraping

Wikipedia

```
# To look at whether we are allowed to do web scraping
paths_allowed("https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago")
```

```
##  en.wikipedia.org
```

```
## [1] TRUE
```

```r
# give URL and read HTML
url <- read_html("https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago")
str(url)
```

```
## List of 2
##  $ node:<externalptr>
##  $ doc :<externalptr>
##  - attr(*, "class")= chr [1:2] "xml_document" "xml_node"
```

1. Grab the html elements

```r
#xpath
nds <- html_elements(url, xpath = '//*[contains(concat( " ", @class, " " ), concat( " ", "us-census-pop-

# storing html
tbl <- html_text(nds)
historical_pop <- tbl[5:44] %>% matrix(ncol=4, byrow = TRUE) %>% as.data.frame()

# change the variable names
names(historical_pop) <- tbl[1:4]

# drop the third column
historical_pop <- historical_pop[, c(1, 2, 4)]
```

```r
# by Sungjoo: X path is different (Oakland, Kenwood, Hyde_Park  & Armour_Square, Douglas, Fuller_Park,
#https://en.wikipedia.org/wiki/Armour_Square,_Chicago
#https://en.wikipedia.org/wiki/Douglas,_Chicago
#https://en.wikipedia.org/wiki/Oakland,_Chicago
#https://en.wikipedia.org/wiki/Fuller_Park,_Chicago
#https://en.wikipedia.org/wiki/Washington_Park_(community_area),_Chicago
#https://en.wikipedia.org/wiki/Hyde_Park,_Chicago
#https://en.wikipedia.org/wiki/Kenwood,_Chicago
```

2. Expanding to More Pages

```r
historical_pops <- historical_pop
directions <- c("Oakland,_Chicago", "Kenwood,_Chicago", "Hyde_Park,_Chicago")
population <- data.frame()

# for loop 1 (Oakland, Kenwood, Hyde_Park)
for (i in directions) {
  url2 <- paste0("https://en.wikipedia.org/wiki/", i)
  src2 <- read_html(url2)

  # xpath
  nds2 <- html_elements(src2, xpath = '//*[contains(concat( " ", @class, " " ), concat( " ", "us-census-
  names2 <- html_text(nds2)

  # storing html
  tbl2 <- html_text(nds2)
  population <- tbl2[5:44] %>% matrix(ncol=4, byrow = TRUE) %>% as.data.frame()

  # change the variable names
  names(population) <- tbl2[1:4]

  # drop the third column
  population <- population[, c(2, 4)]

  #part <- data.frame(names2, population)
  historical_pops <- cbind(historical_pops, population)
}
```

3. Scraping and Analyzing Text Data

(1) Creating a corpus

```
#xpath
nds_text <- html_elements(url, xpath = '//p')
str(nds_text)
```

```
## List of 8
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  $ :List of 2
##   ..$ node:<externalptr>
##   ..$ doc :<externalptr>
##   ..- attr(*, "class")= chr "xml_node"
##  - attr(*, "class")= chr "xml_nodeset"
```

```r
description <- html_text(nds_text)
description
```

```
## [1] "\n"
## [2] "Grand Boulevard on the South Side of Chicago, Illinois, is one of the city’s Community Areas. Th
## [3] "This is one of the two community areas that encompass the Bronzeville neighborhood, with the oth
## [4] "The Harold Washington Cultural Center is one of its newer and more famous buildings. It arose on
## [5] "According to a 2018 US Census American Community Survey, there were 22,784 people and 10,383 hou
## [6] "Grand Boulevard is part of City of Chicago School District #299 and City Colleges of Chicago Dis
## [7] "The Chicago Transit Authority operates the Chicago \"L\" system in the Grand Boulevard community
## [8] "The Grand Boulevard community area has supported the Democratic Party in the past two presidenti
```

```r
description <- description %>% paste(collapse = ' ')
```

(2) Grab the descriptions of the various communities areas using for loop

```r
# creating a corpus
directions <- c("Grand_Boulevard,_Chicago", "Oakland,_Chicago", "Kenwood,_Chicago", "Hyde_Park,_Chicago
descriptions <- data.frame()

for (i in directions) {
  url2 <- paste0("https://en.wikipedia.org/wiki/", i)
  src2 <- read_html(url2)

  #xpath
  nds_text <- html_elements(src2, xpath = '//p')

  description <- html_text(nds_text)
  description <- description %>% paste(collapse = ' ') %>%
    as.data.frame()

  descriptions <- rbind(descriptions, description)
}
```

(3) Create tokens using unnest tokens. Make sure the data is in one token per row format. Remove any stop words within the data. What are the most common words used overall?

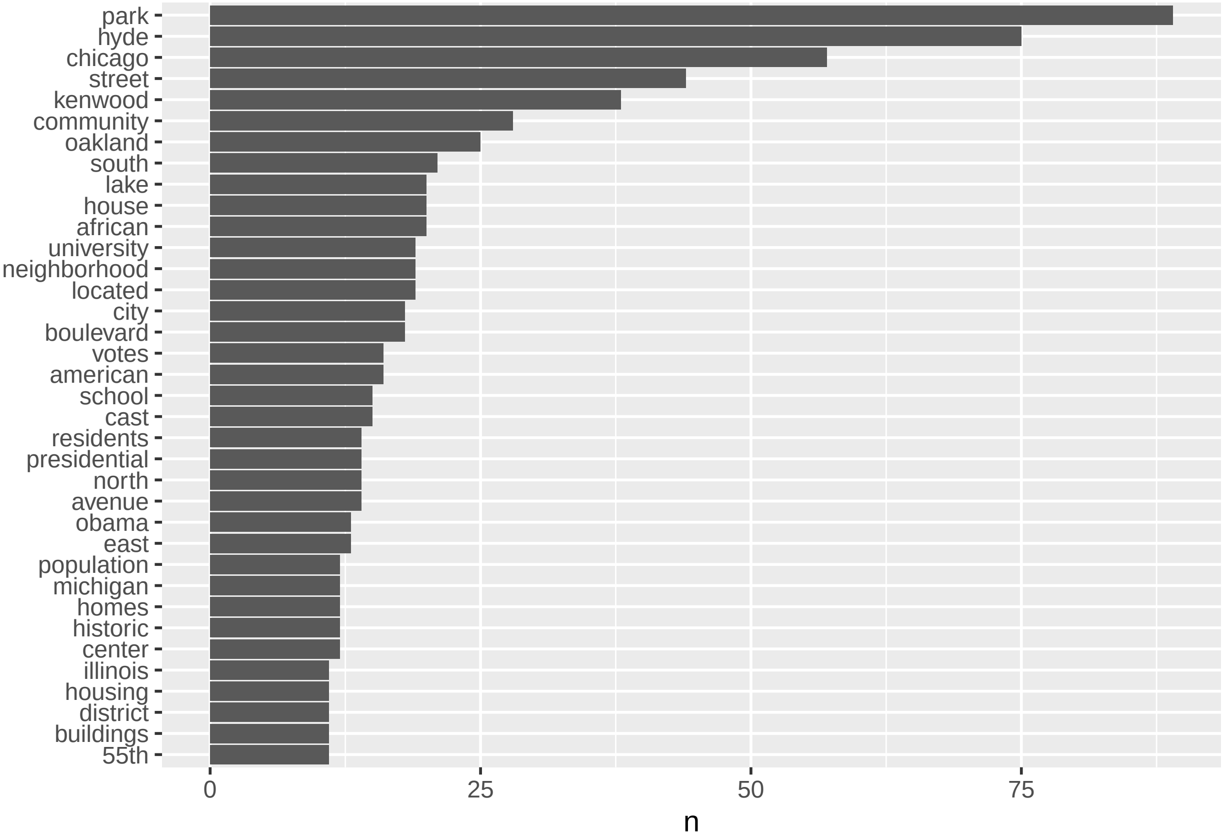
The most common words used overall is ‘park’ (n=89).

```r
# create tokens using unnest_tokens
text_df <- tibble(location = c("Grand_Boulevard", "Oakland", "Kenwood", "Hyde_Park"), text = descriptio


token <- text_df %>%
  unnest_tokens(word, text)
```

```r
# remove stop words
data("stop_words")
token <- token %>%
  anti_join(stop_words)
```

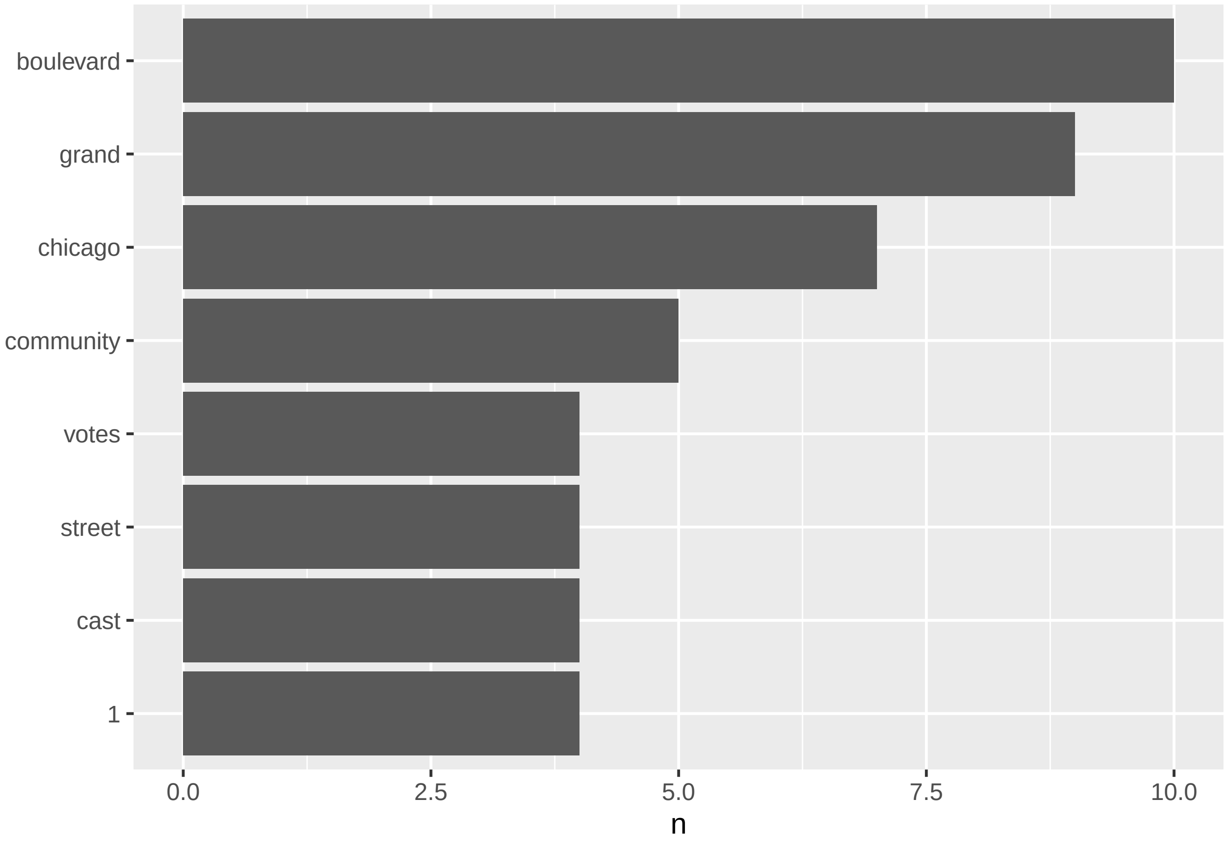
```
## Joining with `by = join_by(word)`
```

```r
# find the most common words in all the books as a whole
token_count <- token %>%
  count(word, sort = TRUE)

# plot the most common words overall
token %>%
  count(word, sort = TRUE) %>%
  filter(n > 10) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)
```

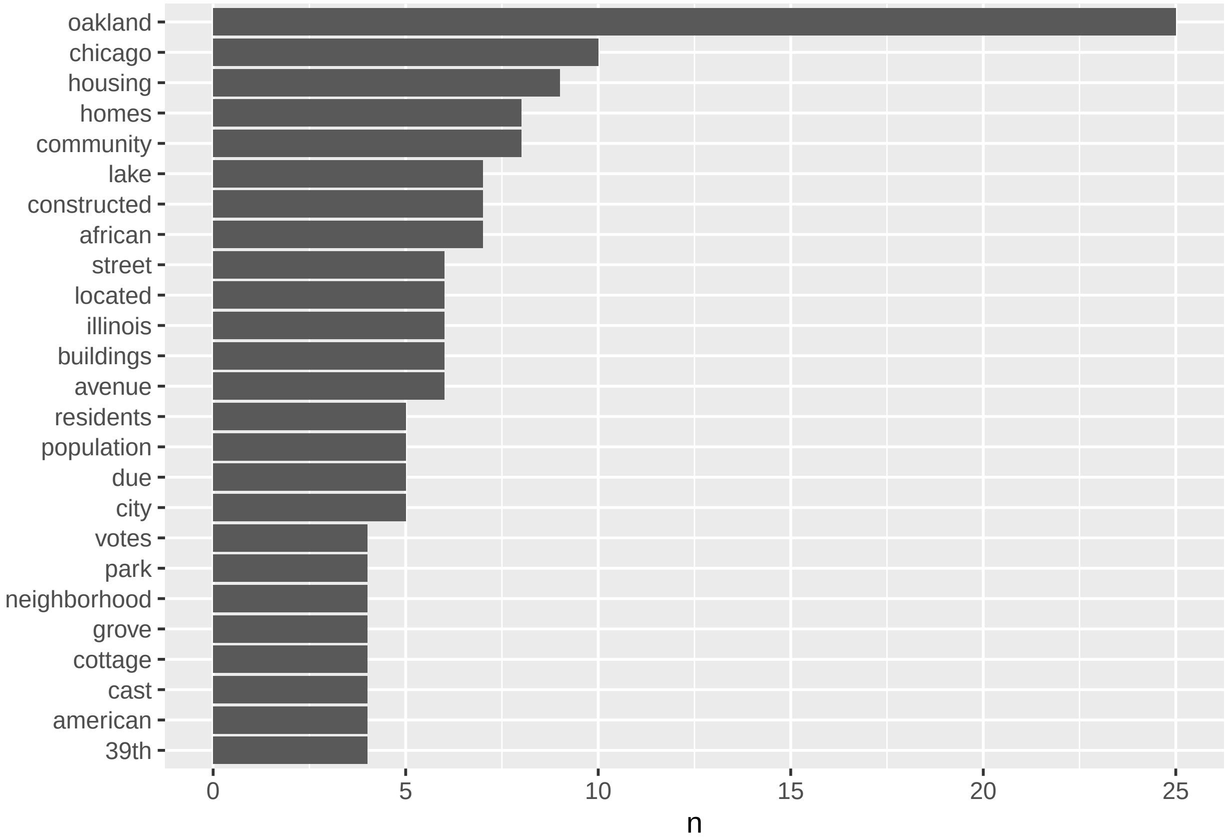


(4) Plot the most common words within each location. What are some of the similarities between the locations? What are some of the differences?

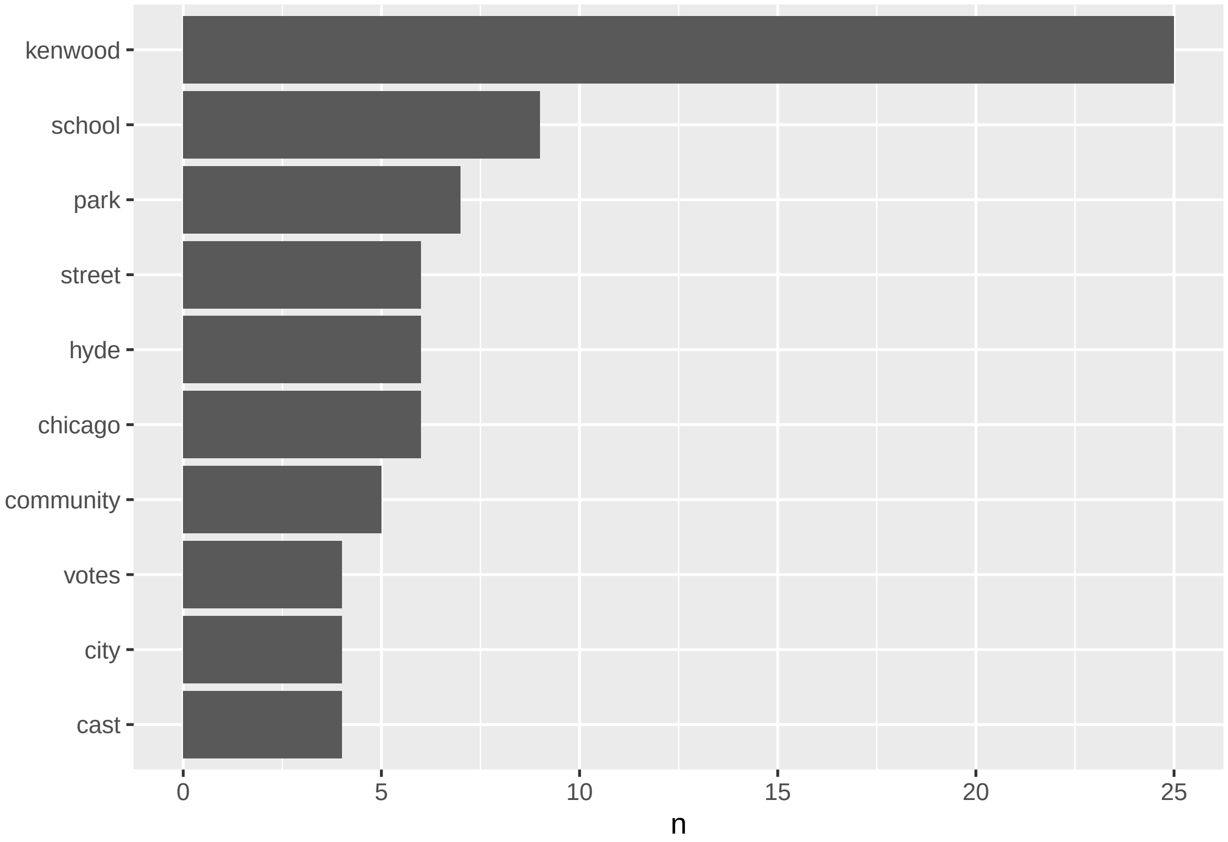
```r
# plot the most common words within Grand_Boulevard
token_Grand_Boulevard <- token %>%
  filter(location == "Grand_Boulevard") %>%
  count(word, sort = TRUE) %>%
  filter(n > 3) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)

# plot the most common words within Oakland
token_Oakland <- token %>%
  filter(location == "Oakland") %>%
  count(word, sort = TRUE) %>%
  filter(n > 3) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)

# plot the most common words within Kenwood
token_Kenwood <- token %>%
  filter(location == "Kenwood") %>%
  count(word, sort = TRUE) %>%
  filter(n > 3) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)

# plot the most common words within Hyde_Park
token_Hyde_Park <- token %>%
  filter(location == "Hyde_Park") %>%
  count(word, sort = TRUE) %>%
  filter(n > 5) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)

par(mfrow=c(2,2))
token_Grand_Boulevard
```



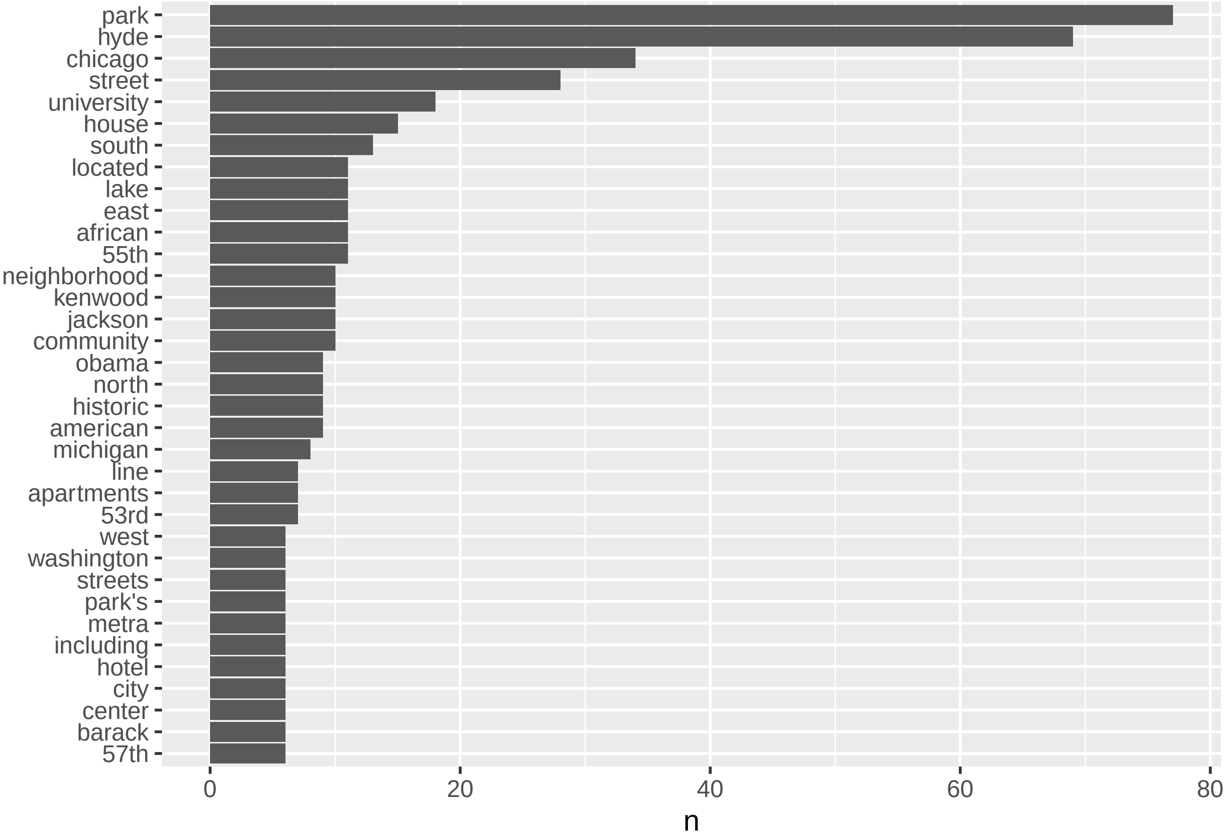
```
token_Oakland
```



```
token_Kenwood
```



```
token_Hyde_Park
```


park
hyde
chicago
street
university
house
south
located
lake
east
african
55th
neighborhood
kenwood
jackson
community
obama
north
historic
american
michigan
line
apartments
53rd
west
washington
streets
park's
metra
including
hotel
city
center
barack
57th
0
20
40
60
80
n