R in Real Life:

EVIDENCE FROM KELLI'S JOB MARKET PAPER

Context

- **Data:** digitized clinical texts
- **Method:** Text Analysis & Natural Language Processing (NLP)

- *But how do I implement this in practice?*

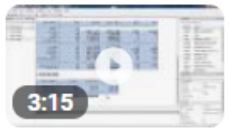
how to do NLP in Stata

All | Videos | News | Images | Shopping | More

About 25,500 results (0.40 seconds)

www.stata.com › how-do-you-stata

How do you Stata? - Stata



How **do** you **Stata**? Point-and-click • » Type • » Program ». Statistical software any way you want it. Whatever ...
Oct 16, 2015 · Uploaded by StataCorp LLC

www.stata.com › features › overview › nonparametric-reg...

Nonparametric regression | Stata

In nonparametric regression, you **do** not specify the functional form. You specify the dependent variable—the ...
Jun 6, 2017 · Uploaded by StataCorp LLC

www.stata.com › features › overview › long-strings

Long strings | Stata

You **can** now use **Stata's** string variables to hold exceedingly long strings, even the contents of files, and ...
Apr 25, 2015 · Uploaded by StataCorp LLC

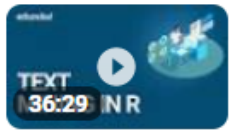
how to do NLP in R

All | Videos | Images | News | Shopping | More

About 357,000 results (0.42 seconds)

www.youtube.com › watch

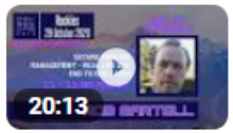
Text Mining In R | Natural Language Processing | Data ...



The following professionals **can** go for this course: 1. ... Text Mining In **R** | **Natural Language Processing** ...
Jan 23, 2019 · Uploaded by edureka!

sweetcode.io › how-to-text-mine-in-r-using-nlp-techniques

How to Text mine in R using NLP techniques – Sweetcode.io



This is where **Natural Language Processing** (**NLP**) **can** enter to tackle the problem, and **R**, the statistical ...
Aug 28, 2018 · Uploaded by Sweetcode HQ

www.youtube.com › watch

R and OpenNLP for Natural Language Processing NLP - Part ...

Overview and demo of using Apache OpenNLP library in **R** to **perform** basic **Natural Language Processing** ...
Apr 30, 2016 · Uploaded by Melvin L

Learn by Example!

- ***Text Mining with R*** by Julia Silge and David Robinson

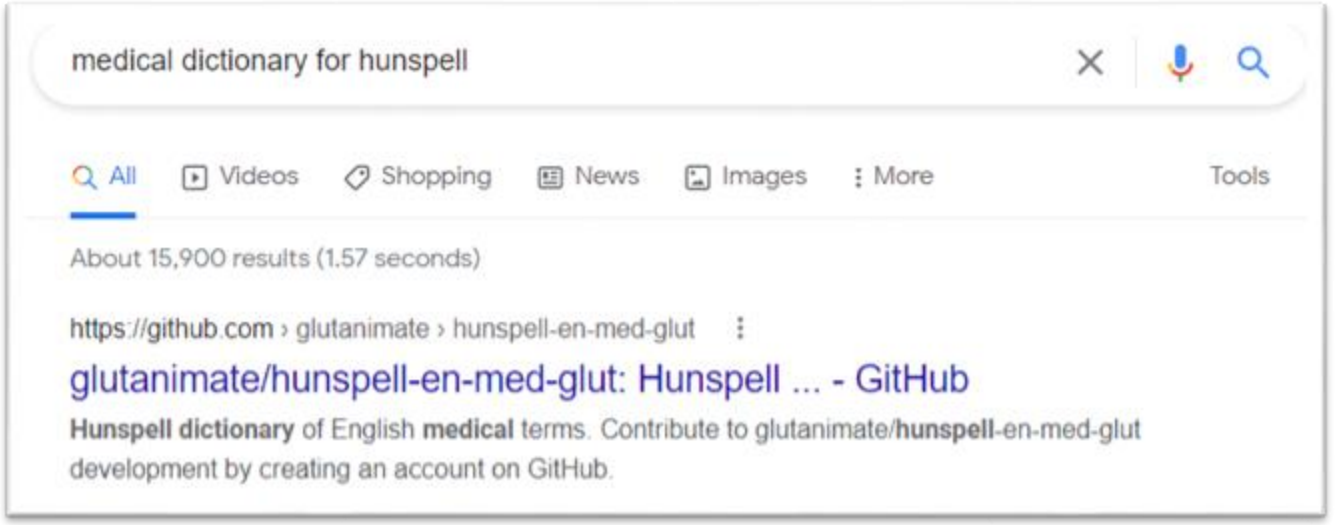
  www.tidytextmining.com

Obstacles along the way- 1

- **Spell check doesn't recognize medical words/abbreviation!**
- "Hunspell" Package

```
➢> hunspell_check("ADHD")
➢[1] FALSE
➢> hunspell_suggest("ADHD")
➢[[1]] [1] "ADD" "ADHARA"
```

- Fix?




```
➢ hunspell_check("ADHD")
➢ [1] TRUE
➢ > hunspell_suggest("ADHD")
➢ [[1]] [1] "ADHD" "ADD"
```

Obstacles along the way- 2

- **I want to remove "stop words" but keep negation!**
- Traditional text cleaning code

```
my_data %>%
  anti_join(stop_words) #will keep all words in my_data that are not in stop_words
```

  - but **stop_words** include "no", "not" , "doesn't",…

- Fix?
  - "qdap" package contains **negation.words**

```
stop_nonegative=stop_words%>%
  filter(!word%in%negation.words) #creates new stop word list that does not include negation

my_data %>%
  anti_join(stop_nonegative)
```

NLP Example: EconTwitter

1. **Get the data!**
   - Twitter API and “rtweet” package
   - many tutorials on webscraping in general

```
#load required packages
library(tidyverse)
library(tidytext)
library(qdap)
library(wordcloud)


#load in the data
econ_twitter=read.csv("tweets.csv", stringsAsFactors = F)
names(econ_twitter)
```

```
"user_id"    "status_id"  "created_at" "text"
```

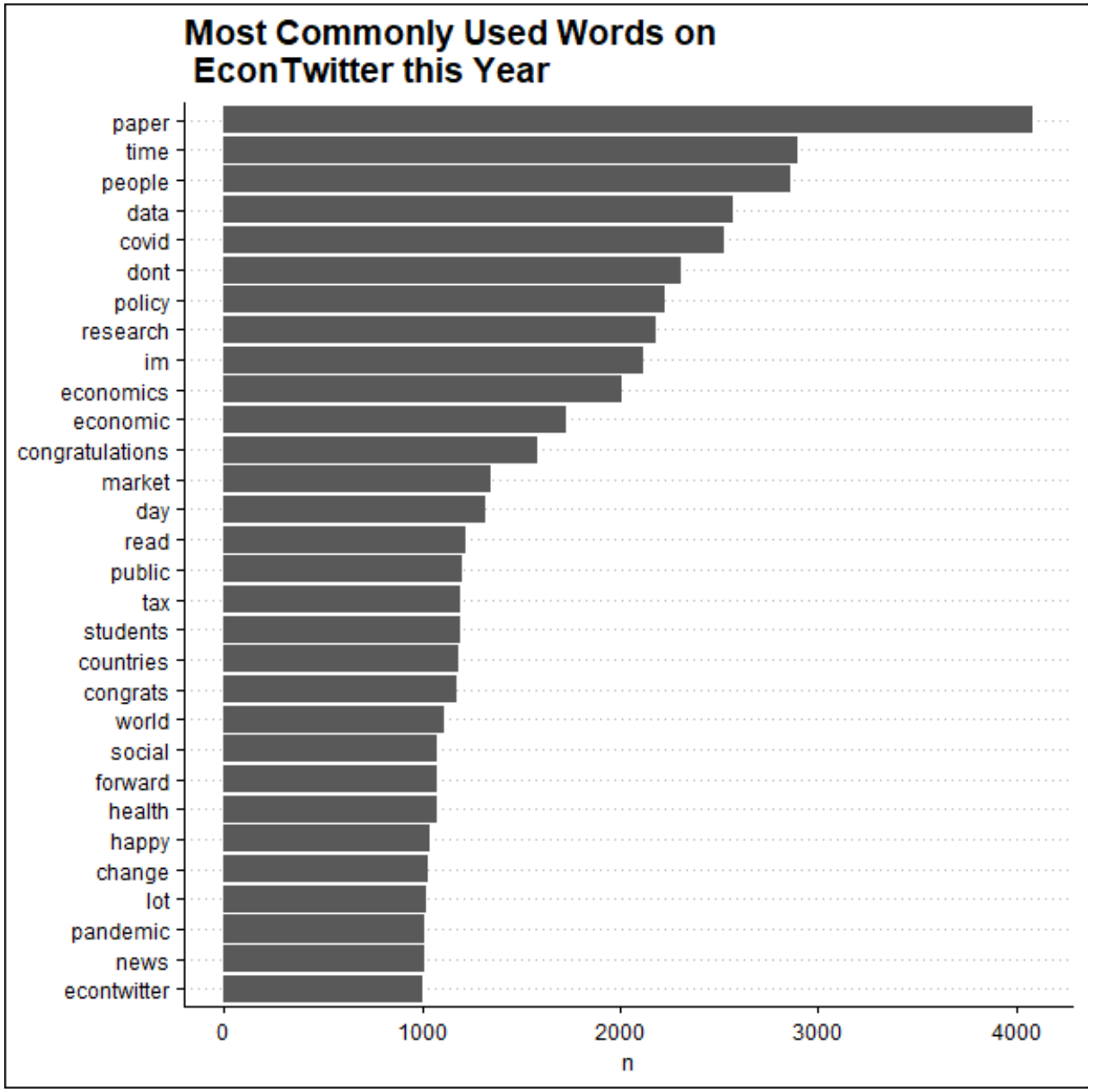
2. **Pre-process the text**
   - Lowercase, remove punctuation, filter stop words, unnest tokens

```
#pre-process the text
  #lowercase, remove punctuation, remove numbers
econ_twitter$text=str_to_lower(econ_twitter$text)
econ_twitter$text=str_remove_all(econ_twitter$text, "[:punct:]")
econ_twitter$text=str_remove_all(econ_twitter$text, "[:digit:]")

#unnest and remove stop_words
tweets_tidy <- econ_twitter %>%
  unnest_tokens(word,text) %>%
  anti_join(stop_words) %>%
  filter(word != "rt", word != "https", word != "t.co", word!="amp")
```

3. Word Frequency

```
#count frequency of words
tweets_freq = tweets_tidy %>%
  count(word, sort = TRUE)
head(tweets_freq)

#frequency plot
tweets_freq %>%
  filter(n > 1000) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n)) +
  geom_bar(stat = "identity") +
  xlab(NULL) +
  coord_flip() +
  ggtitle("Most Commonly Used Words
          on  EconTwitter this Year")
```

```
#word cloud
library(viridis) #needed for color palette
tweets_freq %>%
  with(wordcloud(word, n, max.words = 30,
                 colors = plasma(n=15, direction = -1),
                 rot.per=.5))
```



3. Word Frequency

```r
#count frequency of words
tweets_freq = tweets_tidy %>%
  count(word, sort = TRUE)
head(tweets_freq)

#frequency plot
tweets_freq %>%
  filter(n > 1000) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n)) +
  geom_bar(stat = "identity") +
  xlab(NULL) +
  coord_flip() +
  ggtitle("Most Commonly Used Words
          on  EconTwitter this Year")
```

```r
#word cloud
library(viridis) #needed for color palette
tweets_freq %>%
  with(wordcloud(word, n, max.words = 30,
                 colors = plasma(n=15, direction = -1),
                 rot.per=.5))
```

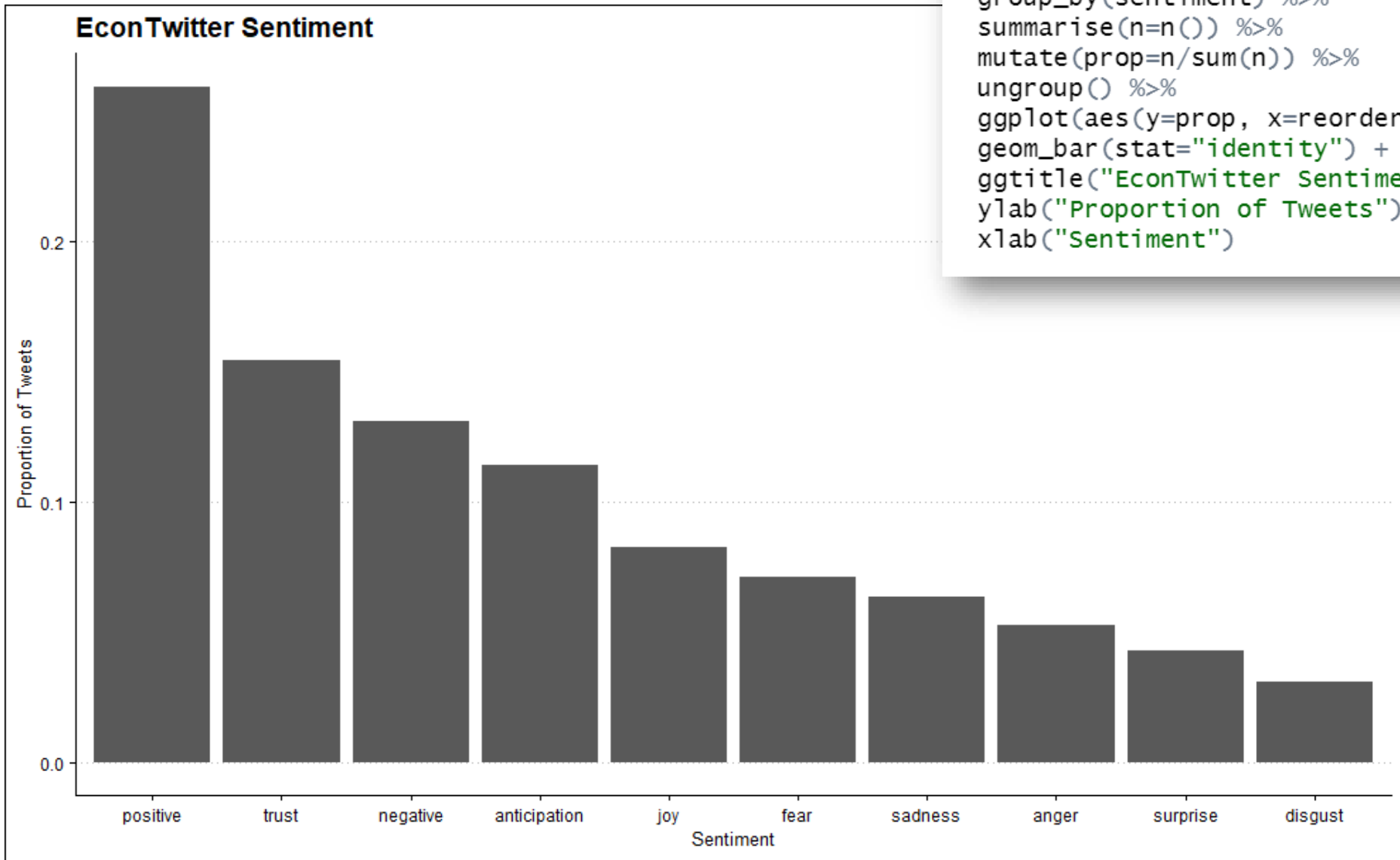
4. Sentiment Analysis

```
#Sentiment Analysis
nrc= get_sentiments("nrc")
head(nrc)
```

```
> head(nrc)
# A tibble: 6 x 2
  word      sentiment
  <chr>     <chr>
1 abacus    trust
2 abandon   fear
3 abandon   negative
4 abandon   sadness
5 abandoned anger
6 abandoned fear
```

4. Sentiment Analysis

```
nrc %>%
  inner_join(tweets_tidy) %>%
  group_by(sentiment) %>%
  summarise(n=n()) %>%
  mutate(prop=n/sum(n)) %>%
  ungroup() %>%
  ggplot(aes(y=prop, x=reorder(sentiment, -prop))) +
  geom_bar(stat="identity") +
  ggtitle("EconTwitter Sentiment") +
  ylab("Proportion of Tweets") +
  xlab("Sentiment")
```



Questions? Comments?

- Kelli.marquardt16@gmail.com