

Project 3 Skill Analysis

Donald Butler

10/15/2021

```
library(tidyverse)
library(tidytext)
library(janeaustenr)
```

```
urlfile<-"https://raw.githubusercontent.com/nolivercuny/data607-team-6-project-3/master/data/job_listing.csv"
jobdat <- read_csv(url(urlfile))
```

```
## Rows: 838 Columns: 13
```

```
## -- Column specification -----
## Delimiter: ","
## chr (11): job_title, company_name, region, salary, employment_type, career_l...
## dbl (2): search_rank, applicant_count
```

```
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
#view short file summary and class
jobdat<-data_frame(jobdat)
```

```
## Warning: 'data_frame()' was deprecated in tibble 1.1.0.
## Please use 'tibble()' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
```

```
glimpse(jobdat)
```

```
## Rows: 838
## Columns: 13
## $ search_rank      <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, ~
## $ job_title        <chr> "Senior Data Scientist", "Data Scientist", "Data Scien~
## $ company_name     <chr> "TextNow", "Amazon", "Alldus", "Facebook", "Google", "~
## $ region           <chr> "New York, NY", "New York, NY", "New York City Metropo~
## $ applicant_count  <dbl> 4, 60, 35, 47, 22, 19, 22, 4, 15, 82, 45, 3, 21, 42, 4~
## $ salary           <chr> NA, NA, NA, "$123,000/yr - $219,000/yr (LinkedIn est.)~
## $ employment_type  <chr> "Full-time", "Full-time", "Full-time", "Full-time", "F~
## $ career_level     <chr> NA, NA, "Entry level", NA, NA, NA, "Mid-Senior level",~
## $ company_size     <chr> "51-200 employees", "10,001+ employees", "11-50 employ~
```

```
## $ industry      <chr> "Telecommunications", "Internet", "Staffing & Recruitin~
## $ date_queried  <chr> "10/12/21 22:10", "10/12/21 22:10", "10/12/21 22:10", ~
## $ date_posted   <chr> "10 hours ago", "2 weeks ago", "3 weeks ago", "1 week ~
## $ description   <chr> "TextNow is based around a simple idea: Communication ~
```

Created ngrams of 1, 2, and 3 words, filtered out the common stop_words, and then did a count of each and filtered down to the rows that occurred at least 10 times within our description dataset. I then exported the results which contained 6015 rows to a file which I loaded into Excel to determine which were relevant job skills and which were not.

```
jobdat_1gram <- jobdat %>%
  unnest_tokens(ngram,description,token='ngrams',n=1,format='text',drop=TRUE,to_lower=TRUE) %>%
  filter(!ngram %in% stop_words$word) %>%
  count(ngram,sort = TRUE) %>%
  filter(n >= 10)

jobdat_2gram <- jobdat %>%
  unnest_tokens(ngram,description,token='ngrams',n=2,format='text',drop=TRUE,to_lower=TRUE) %>%
  separate(ngram,c('word1','word2'),sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  unite(ngram, c('word1','word2'), sep = " ") %>%
  count(ngram,sort = TRUE) %>%
  filter(n >= 10)

jobdat_3gram <- jobdat %>%
  unnest_tokens(ngram,description,token='ngrams',n=3,format='text',drop=TRUE,to_lower=TRUE) %>%
  separate(ngram,c('word1','word2','word3'),sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(!word3 %in% stop_words$word) %>%
  unite(ngram, c('word1','word2','word3'), sep = " ") %>%
  count(ngram,sort = TRUE) %>%
  filter(n >= 10)

jobdat_ngrams <- jobdat_1gram %>%
  rbind(jobdat_2gram) %>%
  rbind(jobdat_3gram) %>%
  arrange(desc(n))

head(jobdat_ngrams,25)
```

```
## # A tibble: 25 x 2
##   ngram          n
##   <chr>        <int>
## 1 data          8429
## 2 experience    3474
## 3 business     2797
## 4 team         2346
## 5 science      2039
## 6 learning     2003
## 7 machine      1440
## 8 product      1416
```

```
## 9 machine learning 1391
## 10 models 1305
## # ... with 15 more rows
```

```
jobdat_ngrams %>%
  write.table(file = './jobdat_ngrams.csv', quote = FALSE, sep = '\t', row.names = FALSE)
```

I generated a file of applicable skills, along with common alternative spellings, ML & Machine Learning. On lines with multiple alternatives, they are separated by a pipe |, for RegEx comparison.

```
(jobskills <- read_csv('https://raw.githubusercontent.com/dab31415/DATA607/main/Projects/Project_3/JobSkills.csv'))
```

```
## Rows: 53 Columns: 1
```

```
## -- Column specification -----
## Delimiter: ","
## chr (1): JobSkill
```

```
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
## Warning: One or more parsing issues, see 'problems()' for details
```

```
## # A tibble: 53 x 1
##   JobSkill
##   <chr>
## 1 Advanced Analytics
## 2 Agile
## 3 AI|Artificial Intelligence
## 4 API|Application Programming Interface
## 5 AWS|Amazon Web Services
## 6 Azure
## 7 Bayesian
## 8 Bioinformatics
## 9 Biology
## 10 Cloud Computing
## # ... with 43 more rows
```

I loop through the list of skills and use regex to determine if the skill is listed within the description text. A new attribute is created in the jobs data frame that indicates if the skill is required.

```
for (i in 1:nrow(jobskills)) {
  jobdat[,ncol(jobdat) + 1] <- str_detect(jobdat$description, regex(paste('[^A-Z0-9]',jobskills[i,1]), '[^A-Z0-9]'))
  colnames(jobdat)[ncol(jobdat)] <- as.character(jobskills[i,1])
}

glimpse(jobdat)
```

```
## Rows: 838
## Columns: 66
```

## \$ search_rank	<dbl> 0, 1, 2, 3, 4, 5, 6, 7, ~
## \$ job_title	<chr> "Senior Data Scientist",~
## \$ company_name	<chr> "TextNow", "Amazon", "Al~
## \$ region	<chr> "New York, NY", "New Yor~
## \$ applicant_count	<dbl> 4, 60, 35, 47, 22, 19, 2~
## \$ salary	<chr> NA, NA, NA, "\$123,000/yr~
## \$ employment_type	<chr> "Full-time", "Full-time"~
## \$ career_level	<chr> NA, NA, "Entry level", N~
## \$ company_size	<chr> "51-200 employees", "10,~
## \$ industry	<chr> "Telecommunications", "I~
## \$ date_queried	<chr> "10/12/21 22:10", "10/12~
## \$ date_posted	<chr> "10 hours ago", "2 weeks~
## \$ description	<chr> "TextNow is based around~
## \$ 'Advanced Analytics'	<lgl> TRUE, FALSE, FALSE, FALS~
## \$ Agile	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'AI Artificial Intelligence'	<lgl> FALSE, TRUE, FALSE, FALS~
## \$ 'API Application Programming Interface'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'AWS Amazon Web Services'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Azure	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Bayesian	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Bioinformatics	<lgl> FALSE, FALSE, TRUE, FALS~
## \$ Biology	<lgl> FALSE, FALSE, TRUE, FALS~
## \$ 'Cloud Computing'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'CS Computer Science'	<lgl> TRUE, TRUE, TRUE, FALSE,~
## \$ 'Critical Thinking'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Data Analysis'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Data Mining'	<lgl> FALSE, FALSE, FALSE, TRU~
## \$ 'Data Modeling'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Data Structures'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Data Visualization'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Data Warehouse'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Database Databases'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Decision Trees'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Deep Learning'	<lgl> FALSE, TRUE, FALSE, FALS~
## \$ 'ETL Extact, Transform, and Load'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Excel	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Git Github'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Google Cloud'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'IOT Internet of Things'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Java	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ JavaScript	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Jupyter	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'NLP Language Processing Language Processing'	<lgl> TRUE, TRUE, FALSE, FALSE~
## \$ 'Linux Unix'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Logistic Regression'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'ML Machine Learning'	<lgl> TRUE, TRUE, TRUE, FALSE,~
## \$ 'Math Mathematics'	<lgl> TRUE, TRUE, FALSE, TRUE,~
## \$ Matlab	<lgl> FALSE, TRUE, FALSE, FALS~
## \$ 'Models Modeling'	<lgl> TRUE, TRUE, TRUE, FALSE,~
## \$ MySQL	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ noSQL	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Numpy	<lgl> FALSE, FALSE, TRUE, FALS~
## \$ Pandas	<lgl> FALSE, FALSE, TRUE, FALS~
## \$ Perl	<lgl> FALSE, FALSE, FALSE, FAL~

## \$ Physics	<lgl> TRUE, FALSE, FALSE, FALS~
## \$ Postgres	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Power BI'	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ PowerPoint	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ Python	<lgl> TRUE, TRUE, TRUE, TRUE, ~
## \$ R	<lgl> FALSE, TRUE, TRUE, TRUE,~
## \$ SAS	<lgl> FALSE, TRUE, FALSE, FALS~
## \$ Scipy	<lgl> FALSE, FALSE, TRUE, FALS~
## \$ SQL	<lgl> TRUE, TRUE, TRUE, TRUE, ~
## \$ 'Statistics Statistical'	<lgl> FALSE, TRUE, TRUE, TRUE,~
## \$ Tableau	<lgl> FALSE, FALSE, FALSE, FAL~
## \$ 'Time Series'	<lgl> FALSE, FALSE, FALSE, FAL~