Text Analysis with Newspapers, Part 2

Alex Leslie

March 13, 2018

Contents

Welcome! In this workshop we'll be exploring several different techniques for analyzing search results from databases of digitized periodicals: distribution over time, page location, collocates, and uniqueness. We'll be focusing on data from Chronicling America in particular, but these techniques and methods are applicable to any equivalently-structured data from any other database of digitized periodicals.

We'll begin right where we left off at the end of Part 1 of this workshop, with the results of a fuzzy string search. If you attended Part 1 (or completed it on your own), you saved your results as a .csv file; locate that file now and deposit it into R's working directory. If you didn't attend Part 1, I've given you a few .csv files to choose from. Read it in to memory as `hits` by changing the file name in this line of code.

```r
hits <- read.csv("spanish_american_war_sn85035720.csv") %>%
  select(-X)
  # R adds an extra numeric identifier column when reading a .csv; this removes it
```

What we have is a data frame of our search hits, in which each row corresponds to a single hit and each column contains a different piece of information about that hit. Most are metadata: the year, month, day, page number, and total pages of the issue in which the hit occured. The final column is a string of the twenty collocate words immediately preceding and following the hit.

LCCN	Year	Month	Day	Page	Issue_Length
sn85035720	1903	5	14	seq-2	6
sn85035720	1903	6	13	seq-1	6
sn85035720	1903	7	29	seq-1	6

x
was in waiting to watch the president turn his shovelful of earth for the mckinley monument many representatives of the spanish american war veterans the grand ar my and thq fioneers were in attend ance and they were referred to in eulo gistic
printed in the news over two months ago the lottie moore gave the local customs officers considerable concern during the spanish american war when she laid at south amboy with coal bound for southern ports she was never heard from fter leav ing
mr sweet will go to the catskill mountains to recuperate he has not been in very good health since the spanish american war in which he took part as a member of the 71st v regiment of new york during the war he

It's important to emphasize that each hit represents a single use of the search name or phrase, *not* a single article in which that name or phrase occured (which is how search results are usually organized in databases and online generally). If a name is only used once in an article, as is often the case, this doesn't make any difference. Longer, more focused articles have the potential to generate multiple separate hits.[1] But this is actually better for most purposes: data organized by the total number of occurrences rather than the number of articles better reflects the amount of space devoted to the particular person, event, or topic in question.[2]

It should be added that the collocates here are not necessarilly from the article in which a search name or phrase occured: if the desired pattern of characters appeared at the beginning of an article, for example, the first half of the collocates string would actually be from the end of a (potentially totally) different article. (This isn't necessarily a bad thing either: after all, this is how it appeared in print). The more results we have, however, the more we can count on irregularities being filtered out.

Quantifying Hits Over Time

One of the most useful and straightforward forms of quantitative analysis when it comes to working with data from periodicals is distribution of reference over time. First, let's simply make a bar graph of the raw number of references by month. Be sure to change the title of your graph by editing the text in the `ggtitle` function!

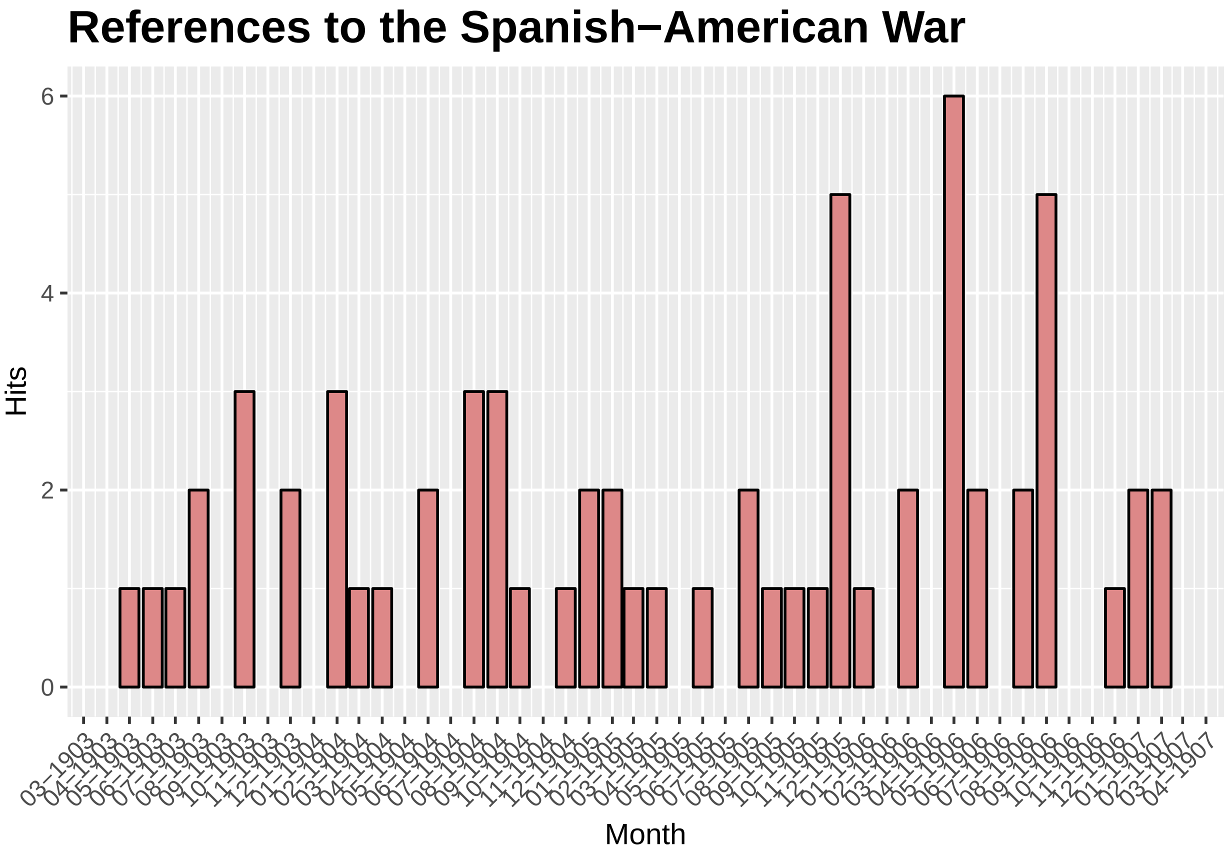
```r
plot <- hits %>%
  group_by(Year, Month) %>%
  summarize(Total=n())
  # group the data by year and month and summarize the number of hits as `Total`

plot$Date <- as.Date(paste(plot$Year, plot$Month, "01", sep="-"), format="%Y-%m-%d")
# R has a special date format that makes generating visualizations much easier

# the next block of code generates a graph, but most lines are purely aesthetic. the first
# four lines do the bulk of the work: selecting the data to be used in the x and y axes
# (lines 1-2), specifying a bar graph (line 3), and scaling the x axis by months (line 4).
plot %>%
  ggplot(aes(x=Date, y=Total)) +
  geom_bar(colour="black", fill="#DD8888", stat="identity") +
  scale_x_date(date_breaks = "month", date_labels=("%m-%Y")) +
  theme(legend.position="bottom") +
  labs(x="Month", y="Hits") +
  ggtitle("References to the Spanish-American War") +
  theme(plot.title = element_text(face="bold", size=rel(1.5))) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

[1] If a person, in particular, is only referred to by their surname after the first reference to them, these subsequent uses won't be recognized as hits.

[2] That said, converting the overall number of references into the number of articles containing a reference wouldn't be hard: one would simply collapse all hits on the same page of the same issue into a single entry.



Simply knowing the raw number of references doesn't actually reveal a whole lot, however. What's missing from this analysis is some measure of the significance of each hit. This is especially important when comparing usage in multiple newspapers.

One way to measure significance of reference is frequency. Word frequency - the number of references within a given time span divided by the total number of words over the same period - is one common approach. But as we've already discovered, newspaper OCR data tends to churn out a lot of word fragments. The total number of words is thereby significantly inflated, meaning that any word-based frequency analysis would significantly misrepresent the actual frequency of reference. Measuring frequency by dividing by the total number of pages within a particular period is a more stable approach.

There is a substantial difference in the number of pages the *Perth Amboy Evening News* printed per issue even between 1903 and 1907, the first and last year of our data - yet another reason why the raw number of references is limited. This means that while we have the number of pages in each issue for which there was a search hit, we'll need to go back to our filepaths in order to pick up the exact number of pages printed per month. You'll recognize this code as a slight adaptation of what we used in the previous workshop to generate an input matrix.

```
years <- list.files("/Users/ahl80/Downloads/sn85035720")
for (k in seq_along(years)) {
  months <- list.files(paste("/Users/ahl80/Downloads/sn85035720/", years[k], sep=""))
  for (j in seq_along(months)) {
    date <- as.Date(paste(years[k], months[j], "01", sep="-"), format="%Y-%m-%d")
    # this line and the next are the only two that have really changed; we only need the
    # year and month in date format, so we paste them together instead

    pages <- length(list.files(Sys.glob(paste("/Users/ahl80/Downloads/sn85035720/",
                         years[k], "/", months[j], "/*/*/", sep=""))))
```

```r
    # the length of the list of all files contained within the particular month directory

    one_month <- data.frame(Date = date, Pages = pages)
    if (exists("by_month")) {
      by_month <- rbind(by_month, one_month)
    } else {
      by_month <- one_month
    }
  }
}
```

Before generating another visualization, we'll combine the metadata we just obtained with the simplified hits data used for the previous visualization. You'll want to edit the text in `ggtitle` again as well.

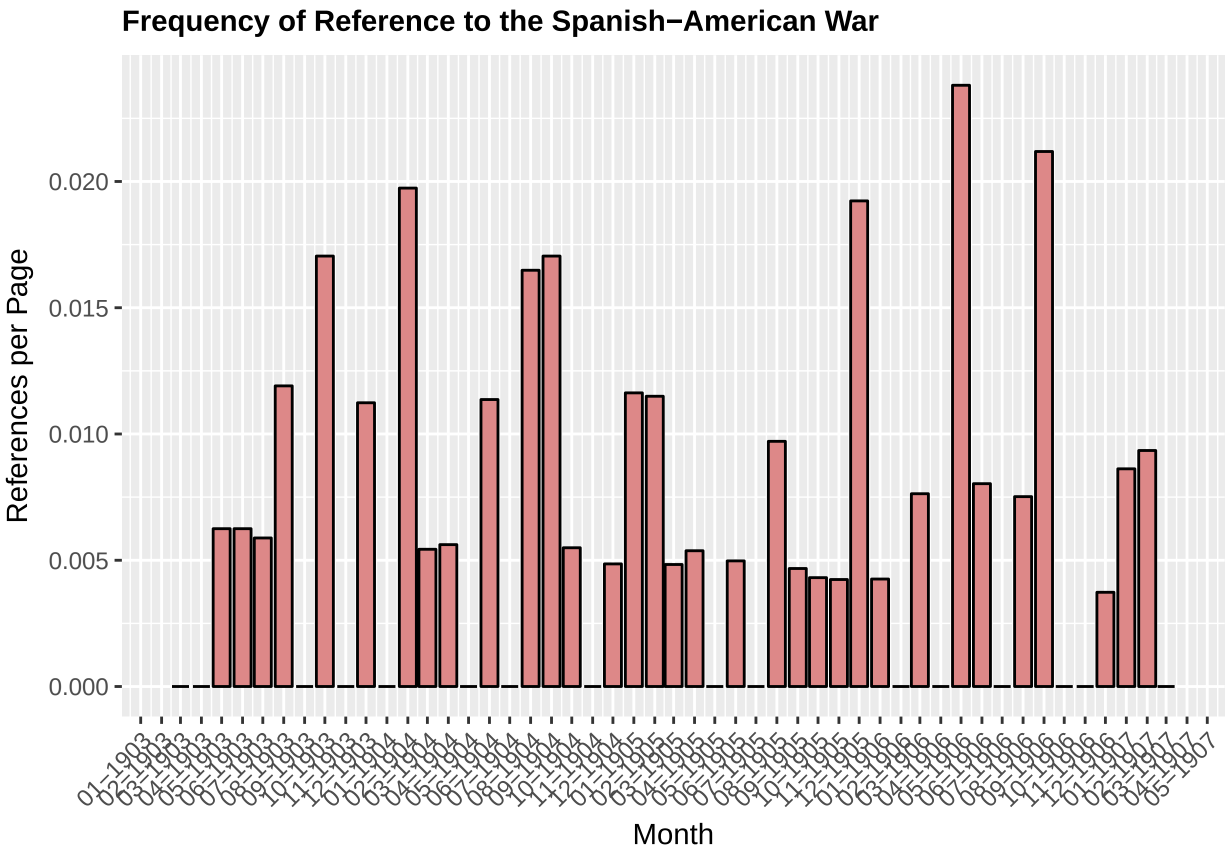
```r
freq <- merge(by_month, plot, by="Date", all=TRUE)
# merge the data frame containing the total number of pages with the data frame of simpli-
# -fied hits data by the Date column, which they share

freq$Total[is.na(freq$Total)] <- 0
# it's necessary to replace null values with 0s when graphing

freq <- freq %>%
  select(Date, Total, Pages)
# getting rid of the columns we don't need

freq %>%
  ggplot(aes(x=Date, y=(Total/Pages))) +
  # the division here is the only real difference between this visualization and the last

  geom_bar(colour="black", fill="#DD8888", stat="identity") +
  scale_x_date(date_breaks = "month", date_labels=("%m-%Y")) +
  theme(legend.position="bottom") +
  labs(x="Month", y="References per Page") +
  ggtitle("Frequency of Reference to the Spanish-American War") +
  theme(plot.title = element_text(face="bold", size=rel(1))) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



This bar graph looks quite similar to the previous at first glance, but there are of course several important differences. What was previously 11 references to the Spanish-American War is now .06 references per page - in other words, the *Perth Amboy Evening News* in March 1904 referenced the Spanish-American War at a rate of about 6% of its pages.[3]

Comparing these two bar graphs reveals another subtle though important difference. (You can do this in RStudio with the back button in the plots viewer.) Though there's something of a spike in the raw number of references to the Spanish-American War in late-1905 and 1906, these months aren't that unusual if we're measuring by frequency: there may be fewer references for 1903 than 1906, but there are also considerably fewer pages in the *Perth Amboy Evening News* in 1903, meaning that those references carry slightly more weight. This would be even more important if we had data back to 1898, when the war was happening but the *Evening News* was probably printing still fewer pages.

In order to compare references across multiple newspapers, we would need to think in some form of frequency in order to make up for differences in issue lengths or numbers. With data from multiple newspapers, we could analyze differences in the amount of attention each gives to a particular person, event or topic. With many newspapers, we could even begin to suggest conclusions about demographics (political, racial, economic, geographic). Rather than moving outward, though, we're going to zoom in and do a bit more finer-grained analysis.

[3] The clunky language here is necessary for precision: remember that our original search returned one result for each approximate match, *not* one result for each page on which there was a match. For this reason, 6% is the rate of reference per page; it is not necessarily the case that 6% of pages contain a reference.

Page Data

Not all references are created equal. Newspapers perform many different cultures functions, and as such they reference the same persons, events, or topics in several different contexts. Fortunately, the internal structure of newspapers generally reflects these contextual differences. In the period we're working with, this structure was organized by page. In order to make sense of page data, we would want to look at some issues of the *Perth Amboy Evening News* itself in a bit more detail. But for sake of time, I'll summarize.

In the late nineteenth and early twentieth centuries, most newspapers were issued either weekly or daily. Daily papers, like the *Perth Amboy Evening News*, had a larger weekend edition (usually on Friday or Saturday) and usually didn't print one day of each week (often Sunday). By the turn of the century, weekly papers and the weekday editions of daily papers were moving towards eight-page issues; weekend editions fluctuated a bit more, between ten- and sixteen-page issues (the difference was largely contingent upon advertising). The format of all these newspapers, however, is actually quite similar, and we can draw several general rules.

The front page generally contains the biggest news stories, and the back page is generally advertisements. The second page often contains state news and comparatively big local news. Page three contains entertainment - fiction, sports, or society news - though in weekend editions one or two of these categories are bumped, often to page seven, eight, or nine. The masthead (as opposed to the nameplate on the front page) usually appears on the left side of page four, regardless of how many pages are in the issue, followed by editorials. The pages following the masthead often contain classifieds and local news listed by town (one page in a six-page issue, separate pages in issues eight pages or more). This gets us up to eight pages. The additional, final pages in weekend editions generally contain a bit more international news and literary content or cultural news, but they are primarily filled with advertisements: it's not uncommon to see a single column surrounded by ads on pages nine and on.[4]

First, how many hits definitely appear in weekend as opposed to weekday editions? In other words, what is the `length` of the vector of hits `which` come from issues with more than eight pages?

```
length(which(hits$Issue_Length > 8))
```

```
## [1] 8
```

Just to jog our memories, that's out of a total of:

```
length(hits$Page)
```

```
## [1] 24
```

One third of the references to Jack London were printed in the weekend edition. Note that weekend editions of the *Perth Amboy Evening News* made up only about one fifth of the total pages printed in a week. There are a couple different avenues of analysis one might pursue here. On the one hand, a large proportion of total references in the weekend edition might suggest that the referent is less locally relevant. On the other hand, the weekend editions of newspapers were more widely read; one might consider weighting references from weekend editions when measuring usage over time in order to reflect this fact.

But let's keep moving forward; next, how many hits appear on each page?

```
hits$Page <- as.numeric(sub("seq-", "", hits$Page))
# to make the data easier to work with, we remove the "seq-" before the page number and
# convert the page number from a character to a numeric value

sort(table(hits$Page), decreasing=TRUE)
```

```
##
##  6  5  7  4  2 15  1  3  8  9
##  5  4  4  3  2  2  1  1  1  1
```

[4] These conventions were not as universal as those used in the four-page newspaper format before the Civil War, but they were nonetheless quite consistent. See Barnhurst, Kevin G. and Nerone, John. *The Form of News.* New York: Guilford Press, 2001.

Let’s look more closely at the collocate strings for the hits on just one page, say, page seven.

```
str_split(hits$Collocates[which(hits$Page==7)], " ")
```

```
## [[1]]
##  [1] "form"       "of"         "an"         "nutobio"    "traphy"    
##  [6] "by"         "s"          "weir"       "mitohell"   "the"       
## [11] "listorical" "accuracy"   "of"         "which"      "nas"       
## [16] "not"        "ret"        "been"       "assailed"   "and"       
## [21] "jaok"       "london"     "s"          "absorbing"  "tale"      
## [26] "tito"       "sea"        "wolf"       "there"      "is"        
## [31] "a"          "variety"    "of"         "diverting"  "aud"       
## [36] "humorous"   "short"      "btories"    "of"         "separate"  
## [41] "interest"   "a"          "second"    
## 
## [[2]]
##  [1] "depots"      "dock"        "yards"       "torpedo"     "boats"      
##  [6] "etc"         "etc"         "probably"    "the"         "most"       
## [11] "conspicuous" "american"    "contributor" "to"          "the"        
## [16] "metro"       "politan"     "lor"         "april"       "is"         
## [21] "lack"        "london"      "the"         "now"         "famous"     
## [26] "author"      "of"          "the"         "call"        "of"         
## [31] "the"         "wild"        "be"          "contributes" "to"         
## [36] "tiiis"       "1"           "isue"        "the"         "first"      
## [41] "part"        "of"          "a"          
## 
## [[3]]
##  [1] "maegrath"   "st"         "elmo"       "augusta"    "evans"     
##  [6] "ia"         "am"         "webster"    "handy"      "dictionary"
## [11] "15"         "ieneil"     "boxes"      "line"       "imported"  
## [16] "boxes"      "worth"      "the"        "sea"        "wolf"      
## [21] "jack"       "london"     "eben"       "ho"         "wen"       
## [26] "irving"     "bacheller"  "the"        "j"          "vebster"   
## [31] "s"          "lrnnary"    "dictionary" "43c"        "25c"       
## [36] "up"         "to"         "1"          "50"         "j"         
## [41] "priced"     "75c"        "to"        
## 
## [[4]]
##  [1] "and"           "we"            "can"           "not"          
##  [5] "j"             "my"            "or"            "borrow"       
##  [9] "enough"        "hooks"         "to"            "go"           
## [13] "round"         "wo"            "cry"           "avaunt"       
## [17] "adventure"     "et"            "we"            "crave"        
## [21] "jack"          "london"        "wr"            "yawn"         
## [25] "1"             "begone"        "introspection" "yet"          
## [29] "we"            "call"          "or"            "henry"        
## [33] "lames"         "away"          "with"          "the"          
## [37] "iroblem"       "novel"         "we"            "shout"        
## [41] "and"           "yet"           "ast"
```

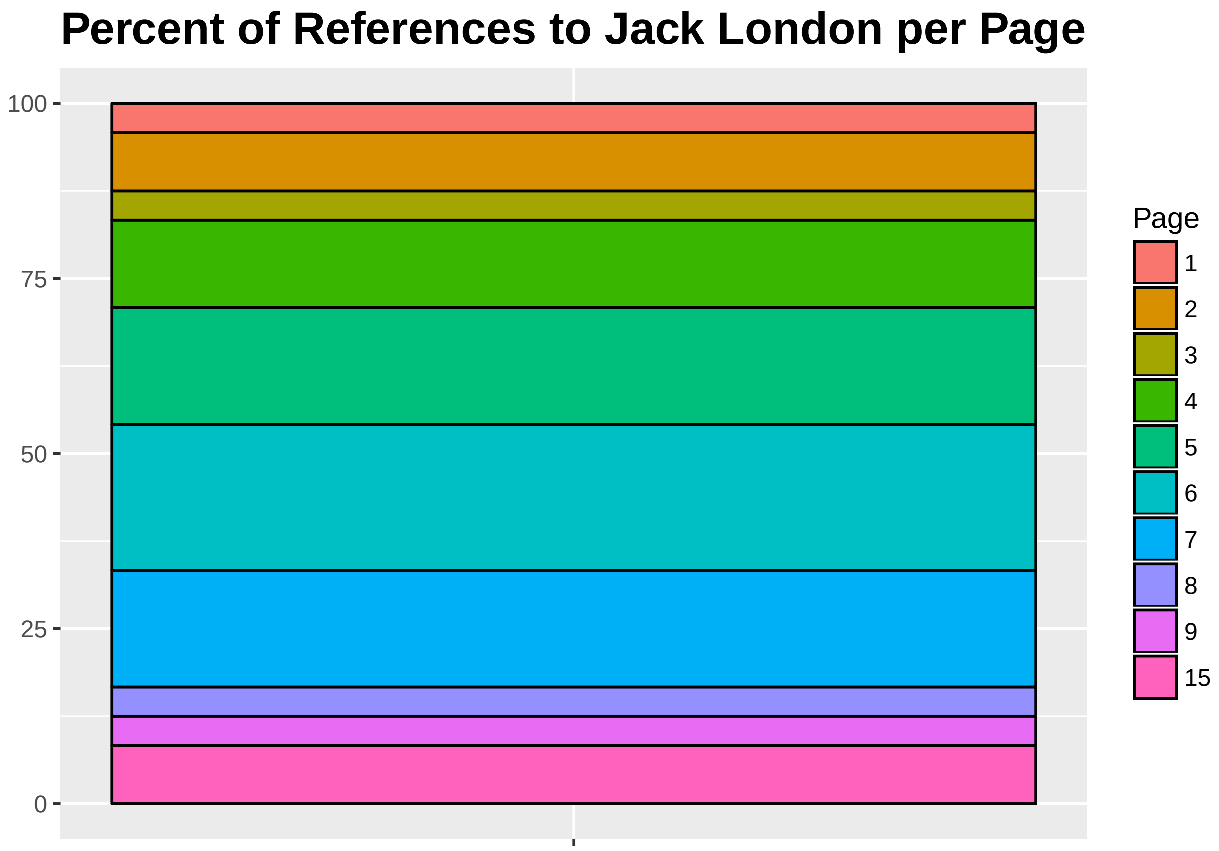
The contexts of these references *should* fit what we would broadly expect for page seven (references to Jack London do at least).

We can generate a visualization to graph what percentage of references to the search phrase appears on each page to further explore page distribution.

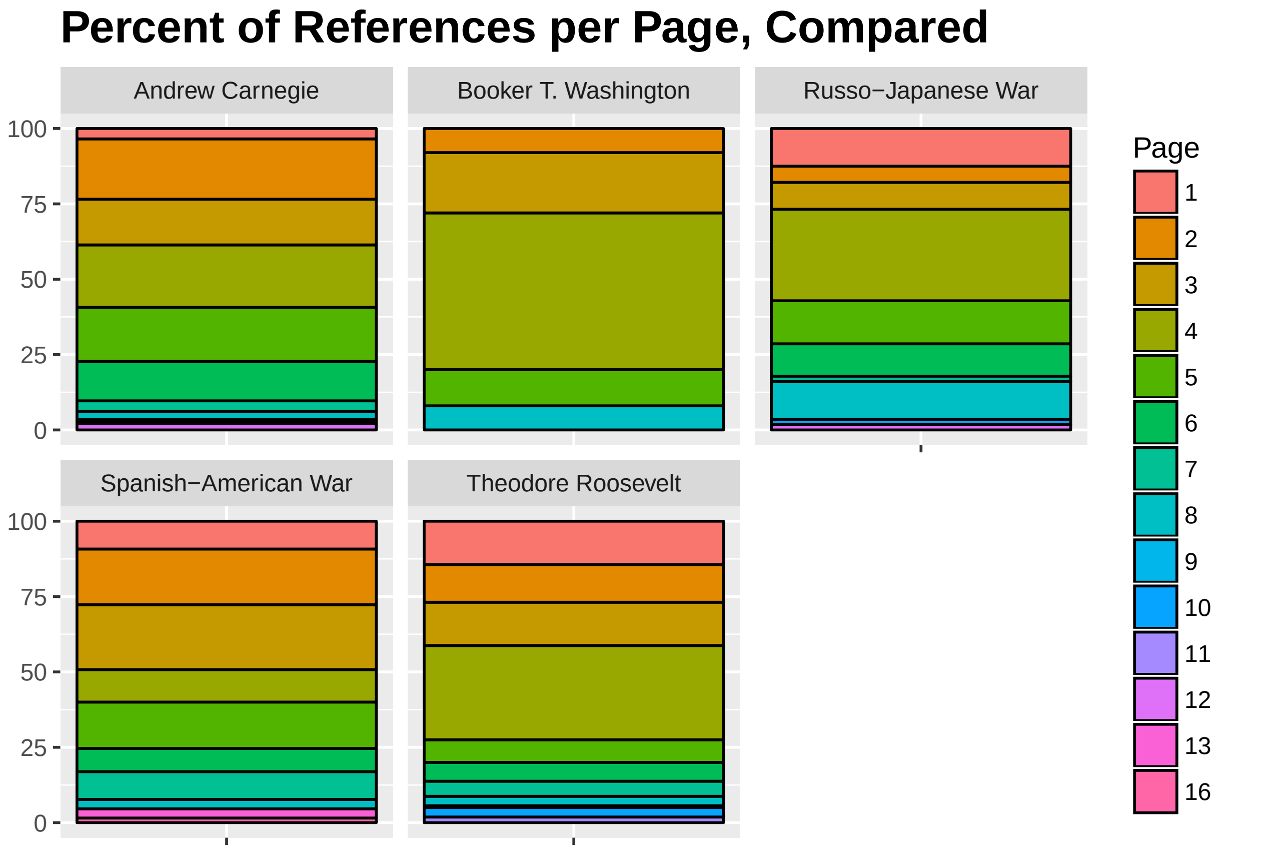
```
graph <- hits %>%
  group_by(Page) %>% summarize(Total=n()) %>%
  mutate(Page_Percents = Total/sum(Total)*100)
  # group the data by page and make a new column `Total` summarizing the total references
  # per page, then make a new column of the % of references per page

graph <- graph %>%
  arrange(Page) %>%
  mutate(Page = as.character(Page))

graph %>%
  ggplot(aes(x="", y=Page_Percents, fill=factor(graph$Page, levels = 1:16))) +
  geom_bar(stat="identity", color="black") +
  theme(axis.title.y=element_blank(), axis.text.x=element_blank()) +
  ggtitle("Percent of References to Jack London per Page") +
  theme(plot.title = element_text(face="bold", size=rel(1.5))) +
  theme(axis.title.x=element_blank()) +
  scale_fill_discrete(name="Page")
```



With a bit of extra fiddling, we can generate a visualization to compare the distribution of references by page for multiple search phrases.



We might begin thinking about this comparison by noting that the Theodore Roosevelt and the Russo-Japanese War, the active president and an ongoing conflict, get referenced much more frequently on the front page and page four, where the biggest news and editorials are found, respectively. Booker T. Washington is referenced more on page four than all other pages combined. A comparatively small proportion of references to Andrew Carnegie are front page news, but a much larger proportion of references to him are found on page two, larger than is the case for most other search phrases; this makes sense, as Carnegie's library gifts in New Jersey and the neighboring states made him newsworthy on a different scale. The Spanish American War, which had been over for five years by the start of the run of the *Perth Amboy Evening News* we're examining, has shifted from more news-focused pages to more culture-focused pages (with more data, we might generate a sequence of graphs to track this shift over time).

Collocates

Page data is useful for inferring broad, structural-level contextual patterns. But more fine-grained sense of usage at the sentence-level is often desirable as well. One accessible means of achieving this is through analysis of collocates, the strings appearing immediately before and after the desired string(s). Conveniently, we grabbed all this information in our original search in the previous workshop.

```
colls <- unlist(strsplit(as.vector(hits$Collocates), "\\W+"))
# these lines organize the collocates into a `table` listing each element of `colls` with
# its number of occurrences, `sort` high to low, and show only the 25 most occurring

sort(table(colls), decreasing=TRUE)[1:15]
```

```
## colls
##      the       of carnegie       to        a   andrew      and      for
##       355       220       174       153       147       130       109        90
##        in         i        is         s       000        by        at
##        90        73        68        67        55        51        50
```

There's quite a lot of static here; selecting only the elements of `colls` that contain more than three characters will remove some of that.

```
top_colls <- colls[which(nchar(colls) > 3)]
sort(table(top_colls), decreasing=TRUE)[1:15]
```

```
## top_colls
##  carnegie    andrew      that      with      from      have      will
##       174       130        47        27        24        21        20
##      been president   library      york      drew      here      said
##        19        19        18        18        15        15        14
##      this
##        14
```

That helped, but there's still too much clutter. This time, we'll use a stop list - a list of common words that we don't want gumming up the works - to cull a bit more.

```
stoplist <- readLines("https://algs4.cs.princeton.edu/35applications/stopwords.txt")
top_colls <- top_colls[-which(top_colls %in% stoplist)]
# exclude (with `-`) all elements in `top_colls` that are also `%in%` the stoplist

sort(table(top_colls), decreasing=TRUE)[1:26]
```

```
## top_colls
##   carnegie     andrew  president    library       york       drew
##        174        130         19         18         18         15
##    college       gift       lias   building       fund     public
##         13         13         12         11         11         11
##       made      march      april     london   received      years
##         10         10          9          9          9          9
##  announced       give       home      state     states      steel
##          8          8          8          8          8          8
##     united university
##          8          8
```

While this gets rid of some real words, it leaves us with more semantically meaningful ones. And indeed, in the case of Andrew Carnegie, this data is interesting: already his steel career has been superseded by his philanthropic career and his union-breaking activities by his altruistic activities. This would not be the case if our dataset was focused fifteen years earlier.

You can probably already imagine the utility of comparative collocates analysis: are the top collocates different in different newspapers (with different policial investments or publishing locations)? Which collocates rise and fall over time? You can also probably already imagine the utility of combining collocates analysis with page location analysis: how do the top collocates differ for different pages? All this is outside the scope of this workshop, but there's one more technique to analyzing newspaper data that I'd like to introduce.

Uniqueness

If not all references are created equal, at least some of them sure look identical. Nineteenth century newspapers regularly reprinted material, whether that material had been written for a previous issue of the same paper, paid for by an advertiser or subscriber, or taken from another newspaper (with or without attribution).

Ascertaining the number of reprints - the degree of uniqueness of reference - is thereby another important means of analyzing broader patterns of relevence.

Since our collocates data is as messy as the newpaper OCR it's drawn from, we can't just run an exact match. You guessed it: it's time to get fuzzy again, this time with `adist`, a function that measures the approximate distance (Levenshtein Distance) between any two strings.

In order to make sure things don't take too long, we'll wrap our code in a function that distributes the workload across multiple processor cores.

```
unique_par <- function (input) {
  core_num <- detectCores()-1
  clust <- makeCluster(core_num, outfile="")
  clusterExport(clust, varlist=c("input"), envir=environment())
  result <- parLapply(clust, seq_along(input$Collocates),
             function (x) {
               if (length(which(adist(input$Collocates[x], input$Collocates) <80)) >1) {
                 "No"
                 } else {
                   "Yes"
                   }
               })
  # this is the line doing the work: for each collocates string, we check `which`, if any,
  # of all the other collocates string is an approximate `adist` match - in this case,
  # within a Levenshtein Distance of 80. If there is one or more match, the function
  # returns "No"; if there are no matches, the function returns "Yes"

  stopCluster(clust)
  return(result)
}
```

We'll assign the results of this uniqueness check to a new column in the `hits` data frame, `Unique`, so that these values remain tied to the hits to which they correspond. This will take a minute or two to run.

```
hits$Unique <- as.character(unique_par(hits))
# this runs our uniqueness function and makes sure the results are characters
```

To check if our data includes any reprints, we'll index the vector of all rows that are not unique into our `hits` data frame.

```
hits[hits$Unique=="No",-7]
```

```
##          LCCN Year Month Day  Page Issue_Length Unique
## 6  sn85035720 1904     3   9 seq-4            6     No
## 7  sn85035720 1904     3  10 seq-4            6     No
## 8  sn85035720 1904     3  11 seq-8           10     No
## 9  sn85035720 1904     3  12 seq-4            6     No
## 10 sn85035720 1904     3  14 seq-4            6     No
## 11 sn85035720 1904     3  15 seq-4            6     No
## 12 sn85035720 1904     3  16 seq-4            6     No
## 13 sn85035720 1904     3  17 seq-4            6     No
## 14 sn85035720 1904     3  18 seq-8           10     No
## 15 sn85035720 1904     3  19 seq-4            8     No
## 16 sn85035720 1904     3  21 seq-4            6     No
## 17 sn85035720 1904     4   2 seq-2            6     No
## 18 sn85035720 1904     4   4 seq-2            6     No
```

```r
kable(hits[hits$Unique=="No",7])
```

x
you only save a nickel harvard lampoon tho north western line russia japan atlas send ton cents in stamps for kusso japanese war alls issued by the t chicago north western it y three lino colored maps each 14x20 bjnnd in convenient form
undertaker then the law yet agatn the north western line boss s japan atlas send ten oents in stamps for rnsso japanese war alls issued by the t chioago nortli western r y three fltia colored maps ottoh 14x30 bound ill convenient form
loan s and tako no sub stitute the north western line russia japan atlas send ton cents in stamps for russo japanese war alls issued by tlie t ohioago north western r y three fine oolorod maps each 14x30 bonnd in convenient form
on electricity mechanics and kindred natters the north western line fuss a japan aline send ten ceuts in stamps for knsso japanese war alls issued by tho t ohioago north western r y three fine colored maps each 14x20 bound in convenient form
england is 210 that of canada 240 the north western line russia japan atlas send ten cents in stamps for russo japanese war alls issued by the t chicago north western r y three fine colored maps enoh 14x20 bonnet in convenient form
cuts bruises sting sprains monarch over pain the north western line russia japan atlas soml ten cents in btamps for ltusso japanese war alls issued by the t chicago nortli western r y throe fine colored maps each 14x20 bound in convenient form
doan s and take tin substitute the north western line huss a japan atlas sond ten cents in stamps for kusso japaneso war alls issncil by tlie t chicago north western r y three flue colored maps each 14x20 bound in convenient form
blood bittcis makes pure blood the north western line puss a japan 1 atlas sond ten cents in stamps for ltnsso japanese war alls issued by tlio t chicago north western r y three fmo colored maps each 14x20 bound ia convenient form
try it sold at sexton s pharmacy the north western line russia japtn atlas send ten cents in stamps for russo japanoso war alls issued by tho t chicago north western r y three fine colored maps each 14x20 bonnd in convenient form
remedy for coughs ami colds the north western line mos a jap in atlas send ten cents in stumps for rufso japanese war alls issuod by the l chicago north western lt y throe tine colored maps each 14x80 boond in convenient form
it bold at sexton s pharmacy he north western line huss a japan atlas send ten cents in stamps for knsso lapancse wnr alls issued by the t chicago north western r y chroe fine colored maps each 14x20 bound in convenient form
jersey fence co mount nolly n j the north western line russia japan atlas send ten oents in stamps for russo japanese war alls issned by the t ohloago north western it y three fine oolored maps each 14x20 bennd in convenient form
jersey fence co mount houy n j the north western line russia japan atlas send ton cents in stamps for kusso japanese war alls issued by the t ohioago nortli western r y throe fine colored maps eaoli 14x20 bound in convenient form

For some searches, there may appear to be no reprinted material in the *Perth Amboy Evening News*. This result is misleading, however, when working with only one newspaper to begin with. It is almost certainly true, to take Jack London as an example, that the *Evening News*‘announcements of magazines’ contents for the month were reprints of the standard announcements found in newspapers across the country. Uniqueness analysis won’t reflect this unless the data includes several newspapers, ideally five or six from the same time span.

This is not the case, however, for references to the Russo-Japanese War. There are nine hits, mostly from the same month, that are clearly all the same reprint - not of news per se but of a particularly topical advertisement.

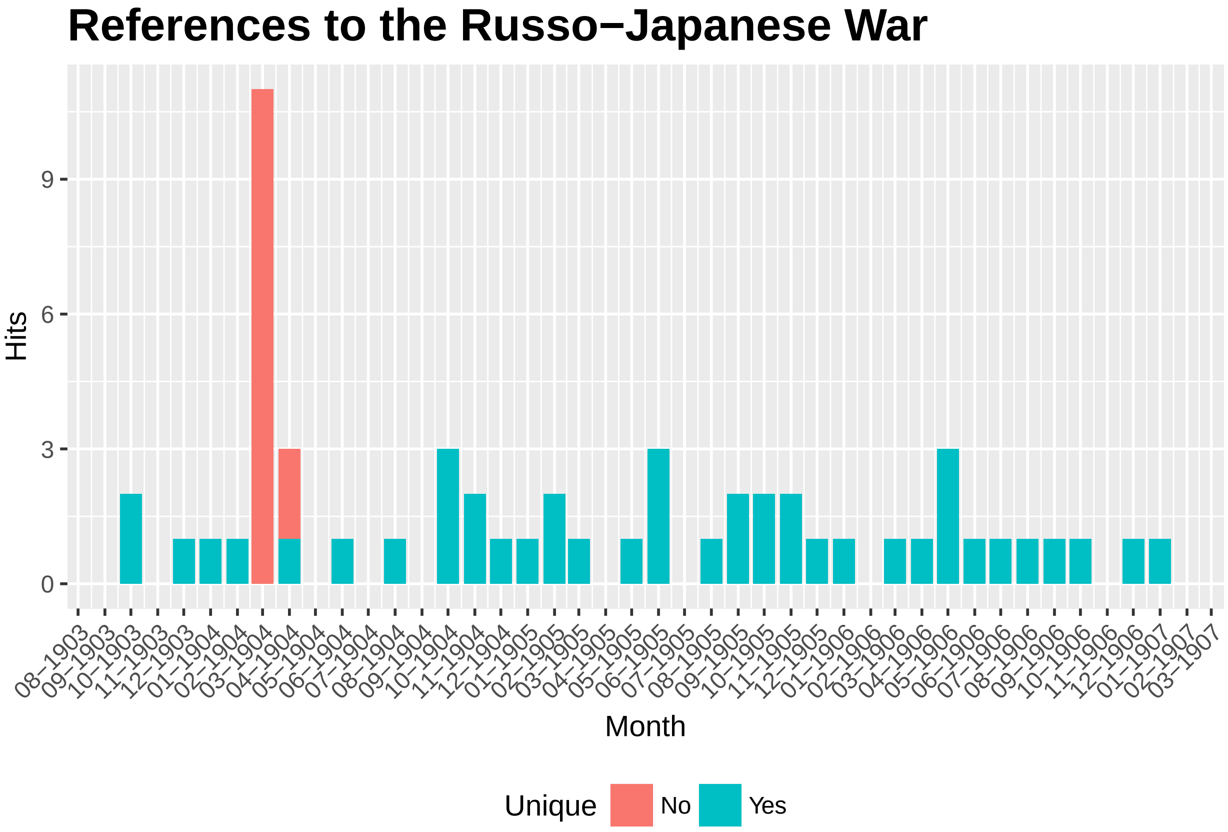
But reprints aren't semantically meaningless![5] They should never be excluded from data; rather, we should try to ascertain what kind of content and conditions cause an article referencing our name or phrase to be reprinted.

To begin to do so, we can graph temporal distribution again with the addition of designating uniqueness as a categorical variable. When working with enough newspapers, this would illustrate the apprxoimate proportion of original references.

```
plot <- hits %>%
  group_by(Unique, Year, Month) %>%
  summarize(Total=n())
plot$Date <- as.Date(paste(plot$Year, plot$Month, "01", sep="-"), format="%Y-%m-%d")

plot %>%
  ggplot(aes(x=Date, y=Total, fill=Unique)) +
  geom_bar(stat="identity") +
  # the `fill` parameter is removed from `geom_bar` and added to `ggplot` to correspond
  # with the `Unique` column of our data frame

  scale_x_date(date_breaks = "month", date_labels=("%m-%Y")) +
  theme(legend.position="bottom") +
  labs(x="Month", y="Hits") +
  ggtitle("References to the Russo-Japanese War") +
  theme(plot.title = element_text(face="bold", size=rel(1.5))) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



[5] See Ryan Cordell and David A. Smith's Viral Texts Project and M. H. Beals' Scissors and Paste Database for two important, ongoing efforts to understand the significance of reprinting in nineteenth-century newspapers. See also McGill, Meredith. *American Literature and the Culture of Reprinting, 1834-1853*. Philadelphia: University of Pennsylvania Press, 2003.

Without distinguishing between reprints and ostensibly non-reprinted material, we might have concluded that the spike in references in March-April 1904 was editorial content or at least news (in the strict sense) about the Russo-Japanese War, which had just begun in February (earlier references are to the "possibility of a Russo-Japanese War"). Instead, advertisements make up a greater deal of the initial hullabaloo, even if they burn out much more quickly (to my knowledge, these reprints are the only advertisements pertaining to the Russo-Japanese War in the *Perth Amboy Evening News*).

If you'd like to keep the additional information we added to your `hits` data frame today, be sure to save it:

```r
write.csv(hits, "name-this-file.csv")
```

Finally, we would really appreciate it if you took a minute to fill out our brief feedback survey.

Thanks for participating!