**The Problem**

The U.S. Defense POW/MIA Accounting Agency maintains POW/MIA data for all our nation’s service members. Matt is working with data [from Korea](http://www.dpaa.mil/Our-Missing/Korean-War/Korean-War-POW-MIA-List/) (the “All US Unaccounted-For” PDF direct link is in the code below) and needed to get the PDF into a usable form and (as you can see if you read through the Twitter thread) both Tabulizer and other tools were introducing sufficient errors that the resultant extracted data was either not complete or trustworthy enough to rely on (hand-checking nearly 8,000 records is not fun).

There PDF in question was pretty uniform, save for the first and last pages. Here’s a sample:

[Click to view slideshow.](https://rud.is/b/2018/07/02/freeing-pdf-data-to-account-for-the-unaccounted/#gallery-10966-1-slideshow)

We just need a reproducible way to extract the data with sufficient veracity to ensure we can use it faithfully.

**The Solution**

We’ll need some packages and the file itself, so let’s get that bit out of the way first:

library(stringi)

library(pdftools)

library(hrbrthemes)

library(ggpomological)

library(tidyverse)

# grab the PDF text

mia\_url <- "http://www.dpaa.mil/portals/85/Documents/KoreaAccounting/pmkor\_una\_all.pdf"

mia\_fil <- "~/Data/pmkor\_una\_all.pdf"

if (!file.exists(mia\_fil)) download.file(mia\_url, mia\_fil)

# read it in

doc <- pdf\_text(mia\_fil)

Let's look at those three example pages:

cat(doc[[1]])

## Defense POW/MIA Accounting Agency

## Personnel Missing - Korea (PMKOR)

## (Reported for ALL Unaccounted For)

## Total Unaccounted: 7,699

## Name Rank/Rate Branch Year State/Territory

## ABBOTT, RICHARD FRANK M/Sgt UNITED STATES ARMY 1950 VERMONT

## ABEL, DONALD RAYMOND Pvt UNITED STATES ARMY 1950 PENNSYLVANIA

## ...

## AKERS, HERBERT DALE Cpl UNITED STATES ARMY 1950 INDIANA

## AKERS, JAMES FRANCIS Cpl UNITED STATES MARINE CORPS 1950 VIRGINIA

cat(doc[[2]])

## Name Rank/Rate Branch Year State/Territory

## AKERS, RICHARD ALLEN 1st Lt UNITED STATES ARMY 1951 PENNSYLVANIA

## AKI, CLARENCE HALONA Sgt UNITED STATES ARMY 1950 HAWAII

...

## AMIDON, DONALD PRENTICE PFC UNITED STATES MARINE CORPS 1950 TEXAS

## AMOS, CHARLES GEARL Cpl UNITED STATES ARMY 1951 NORTH CAROLINA

cat(doc[[length(doc)]])

## Name Rank/Rate Branch Year State/Territory

## ZAVALA, FREDDIE Cpl UNITED STATES ARMY 1951 CALIFORNIA

## ZAWACKI, FRANK JOHN Sgt UNITED STATES ARMY 1950 OHIO

## ...

## ZUVER, ROBERT LEONARD Pfc UNITED STATES ARMY 1950 CALIFORNIA

## ZWILLING, LOUIS JOSEPH Cpl UNITED STATES ARMY 1951 ILLINOIS

## This list of Korean War missing personnel was prepared by the Defense POW/MIA Accounting Agency (DPAA).

## Please visit our web site at http://www.dpaa.mil/Our-Missing/Korean-War-POW-MIA-List/ for updates to this list and other official missing personnel data lists.

## Report Prepared: 06/19/2018 11:25

The poppler library's "layout" mode (which pdftools uses brilliantly) combined with the author of the PDF not being evil will help us make short work of this since:

* there's a uniform header on each page
* the "layout" mode returned uniform per-page, fixed-width columns
* there's no "special column tweaks" that some folks use to make PDFs more readable by humans

There are plenty of comments in the code, so I'll refrain from too much blathering about it, but the general plan is to go through each of the 119 pages and:

* convert the text to lines
* find the header line
* find the column start/end positions from the header on the page (since they are different for each page)
* reading it in with readr::read\_fwf()
* remove headers, preamble and epilogue cruft
* turn it all into one data frame

# we're going to process each page and read\_fwf will complain violently

# when it hits header/footer rows vs data rows and we rly don't need to

# see all those warnings

read\_fwf\_q <- quietly(read\_fwf)

# go through each page

map\_df(doc, ~{

stri\_split\_lines(.x) %>%

flatten\_chr() -> lines # want the lines from each page

# find the header on the page and get the starting locations for each column

keep(lines, stri\_detect\_regex, "^Name") %>%

stri\_locate\_all\_fixed(c("Name", "Rank", "Branch", "Year", "State")) %>%

map(`[`, 1) %>%

flatten\_int() -> starts

# now get the ending locations; cheating and using `NA` for the last column

ends <- c(starts[-1] - 1, NA)

# since each page has a lovely header and poppler's "layout" mode creates

# a surprisingly usable fixed-width table, the core idiom is to find the start/end

# of each column using the header as a canary

cols <- fwf\_positions(starts, ends, col\_names = c("name", "rank", "branch", "year", "state"))

paste0(lines, collapse="\n") %>% # turn it into something read\_fwf() can read

read\_fwf\_q(col\_positions = cols) %>% # read it!

.$result %>% # need to do this b/c of `quietly()`

filter(!is.na(name)) %>% # non-data lines

filter(name != "Name") %>% # remove headers from each page

filter(!stri\_detect\_regex(name, "^(^This|Please|Report)")) # non-data lines (the last pg footer, rly)

}) -> xdf

xdf

## # A tibble: 7,699 x 5

## name rank branch year state

##

## 1 ABBOTT, RICHARD FRANK M/Sgt UNITED STATES ARMY 1950 VERMONT

## 2 ABEL, DONALD RAYMOND Pvt UNITED STATES ARMY 1950 PENNSYLVANIA

## 3 ABELE, FRANCIS HOWARD Sfc UNITED STATES ARMY 1950 CONNECTICUT

## 4 ABELES, GEORGE ELLIS Pvt UNITED STATES ARMY 1950 CALIFORNIA

## 5 ABERCROMBIE, AARON RICHARD 1st Lt UNITED STATES AIR FORCE 1950 ALABAMA

## 6 ABREU, MANUEL Jr. Pfc UNITED STATES ARMY 1950 MASSACHUSETTS

## 7 ACEVEDO, ISAAC Sgt UNITED STATES ARMY 1952 PUERTO RICO

## 8 ACINELLI, BILL JOSEPH Pfc UNITED STATES ARMY 1951 MISSOURI

## 9 ACKLEY, EDWIN FRANCIS Pfc UNITED STATES ARMY 1950 NEW YORK

## 10 ACKLEY, PHILIP WARREN Pfc UNITED STATES ARMY 1950 NEW HAMPSHIRE

## # ... with 7,689 more rows

Now the data is both usable and sobering:

title <- "Defense POW/MIA Accounting Agency Personnel Missing - Korea"

subtitle <- "Reported for ALL Unaccounted For"

caption <- "Source: http://www.dpaa.mil/portals/85/Documents/KoreaAccounting/pmkor\_una\_all.pdf"

mutate(xdf, year = factor(year)) %>%

mutate(branch = stri\_trans\_totitle(branch)) -> xdf

ordr <- count(xdf, branch, sort=TRUE)

mutate(xdf, branch = factor(branch, levels = rev(ordr$branch))) %>%

ggplot(aes(year)) +

geom\_bar(aes(fill = branch), width=0.65) +

scale\_y\_comma(name = "# POW/MIA") +

scale\_fill\_pomological(name=NULL, ) +

labs(x = NULL, title = title, subtitle = subtitle) +

theme\_ipsum\_rc(grid="Y") +

theme(plot.background = element\_rect(fill = "#fffeec", color = "#fffeec")) +

theme(panel.background = element\_rect(fill = "#fffeec", color = "#fffeec"))

