**Analytic Deployment Review of Text.Replace**

Downloading Project:

Using devtools:: install\_github(‘https://github.com/mrkaliv/Text.Replace.git’), I encountered an error that failed the package from being installed. It appeared to be associated with the Namespace.

Error in loadNamespace(j <- i[[1L]], c(lib.loc, .libPaths()), versionCheck = vI[[j]]) :

there is no package called 'quanteda'

Error: Failed to install 'Text.Replace' from GitHub:

(converted from warning) installation of package ‘C:/Users/BENJAM~1/AppData/Local/Temp/RtmpIX38k9/file28bc46dfd97/Text.Replace\_2020.0.1.0.tar.gz’ had non-zero exit status

As an alternative, I opened a new project using Version Control > Git and executed install.packages(‘quanteda’) as the Namespace required three different functions from quanteda. Once that was installed, I installed a new build with no errors.

Documentation Review:

I found a ReadMe document in the inst folder with an overall description of the project. It was not labeled ReadMe, but march\_report, so it was a little confusing that I would find the package description in it. I was expecting to find the proposals completed for class, but understandably, these were artifacts of the class and not specifically documents you would provide in a package. The readme was clear and concise.

Proposal 2 was not located in the package but I was able to find a copy in OPER782.WI2020. I did not understand what was accomplished according to the document. Priority 1 features such as Shiny App had a status of Not Started, while many priority 2 features were in work with a caveat of being coded in python. Display Loaded Data feature was a priority 1 with ‘in work’ status but a caveat that it was paired with the Shiny App feature with ‘maybe’ inclusion in version 1.0.

Text.Replace-package had only the generic help file with no specific information.

Create\_corpus had a help file that assisted with using the function.

Create\_dfm had a help file with a brief description and usage.

Test Project:

Working through the sample\_report readme, survey\_dfm presented warning messages stating that the ngrams argument is not used. At this point, the rest of the readme expressions did not work. Looking back over the create\_dfm help file, I was not sure what to do at this point.

I image that if everything worked, I would end up with a word cloud constructed of terse statements. I could not evaluate the accuracy of the results but seeing how they are being generated by another published function call, I would expect them to be accurate. As stated, I did run into an error while trying to execute the code according to the documentation. However, this may be because it’s still a work in progress as stated by proposal 2. Should everything be in working order, I believe that this package would be easy to use as the examples in the documentation made it clear how to run the functions.

Rubric Assessment:

* Outstanding – ready to publish/deploy (50 pts)
  + All proposal tasks completed
  + Documentation is thorough and clear
  + No errors or warnings during build/use
* Excellent – very little rework required (45 pts)
  + Few proposal tasks incomplete
  + Documentation is clear and mostly complete
  + Minor errors or warnings during build/use
* Satisfactory – extensive rework required (40 pts)
  + Several proposal tasks incomplete
  + Documentation is limited and/or poorly written
  + Severe errors during build or in use
* Unsatisfactory – complete restart required (35 pts)
  + Little progress made on proposal tasks
  + Little to no documentation
  + Cannot be compiled, unusable