The following is a template for COM 314 as to organize work done on data via the 12 Stages.

It is important to realize that while you will cut and paste information from your technical work, you will also have to write explanation of things.

This template will give your headings, wording, and more so it needs to have a system to use.

If words are typed in any format—underlined, italics, all caps, etc.—then you will type those words exactly.

If words are found in brackets [ ], it means you have to add something to what is there to make it your own and focus on your topic and work.

If words are after a hashtag, it will be a note to remember about code, which is the way it works in R too.

If words are in parentheses ( ), it will be like a # but will be about writing the work.

Note: Your page numbers may not match because you may have longer elements of context or explanation; however, everyone’s first page (title page) should look quite similar.

NO brackets, parentheses, or # should appear in the final work.

Remember to stable all pages together and insert page number on the bottom right of each page.

**A Social Media Analysis of COVID-19: Sentiment, Influence, and more on Twitter**

By (Joseph D’Ambrosio)

Introduction: Topic

[write an introduction to your topic that establishes context; likely a paragraph or two and less than 2 pages]

Introduction: Dataset

[explain the size, scope, time, etc. of the dataset you’re using for you topic, how you selected it, and where/how you collected it: likely one to two paragraphs only]

Stage One: Finding average sentiment

[Explain how you found the average sentiment and what the outcome was]

#if you mention sentimentr you need to cite who created or who maintains it

Stage Two: Generating an influence score

[Explain what an influence score is and how it was created for your dataframe]

#if you mention dplyr and the mutate function then you need to cite who created or who maintains it

Stage Three: Individualized sentiment

[Explain why the original sentiment scores needed to be merged into tweets rather than sentences]

Stage Four: Expanding the dataframe to have both influence and sentiment

[Explain how you added the sentiment scores to the dataframe]

Stage Five: Renaming and improving the dataframe

[Explain why changing a column name may be useful; how did you change it?]

Stage Six: Working with time

[Explain why having time as a factor doesn’t work and changing its format to date]

Stage Seven: Tweets across time

[Explain how using the new date column you can visualize the number of tweets per day]

(place the plot you create after the explanation; in addition, you will have to create a label for the plot and center it and the image—this label is NOT the “main =” label you had on the plot itself but rather a new label to integrate it into this written work)

Image 1: Tweets across time

[insert your specific image]

Stage Eight: Examining changes across time

[Explain how to group influence and sentiment by time]

Image 2: Influence on each day

[insert image of your output showing a table of each day and its influence]

#this will look like the image found in stage 8 of the rubric

One can also look at sentiment in the same way.

Image 3: Sentiment on each day

[insert image of your output showing a table of each day and its sentiment]

#this will look like the image found in stage 8 of the rubric

Stage Nine: Visualizing changes over time

One can visualize the outputs noted in images 2 and 3. Plotting each output aids in interpretation. One finds that (finish this sentence).

[what is occurring? Is it increasing steadily? Haphazard? Going down? Is one trait (e.g., influence) going up while the other (e.g., sentiment) going down? Any idea for the cause of the trend?]

Image 4: Influence across time

[insert image of your plot of influence across the days]

Additionally, we can examine visually the sentiment for each day.

Image 5: Sentiment across time

[insert image of your plot of sentiment across the days]

Stage Ten: Influencers and their attributes

[Explain who the top influencers are and the traits you found out about them. Male/female? Organization? How long have they been on Twitter? How often do they tweet? Do all tweet with same pattern of days or times? Etc.]

Stage Eleven: Top positive and negative sentiment

[Explain how you found the top positive and negative user for sentiment. Does the sentiment make sense? Did the sentimentr package make any error in coding their sentiment? Etc.]

Stage Twelve: Looking for relationships

One way to look for relationships between variables is to run a correlation to see the strength and direction of their relationship. Given that the dateframe has a number of numeric values, it would be inefficient take each individually and compare individually. To save time, we will run a correlation plot matrix, which will allow us to compare all numeric values to each other simultaneously. To begin, we condense the original dataframe called [insert your df name] into a dataframe called [insert your df name] that has removed all column that are not numeric. We, then, create a correlation plot matrix, and its outcome can be seen in Image 6.

Image 6: Correlation plot matrix for [df name]

[insert image of your correlation plot matrix which should look similar to the one in the rubric]

[Explain which traits or variables have the strongest correlation? Which have the weakest]

Stage Thirteen: Advanced sentiment of top influencers

[Explain and show how you used advanced sentiment on the top influencer tweets; what do you find? Is it what you expected or something different? Is there any clear reason for it?]

Stage Fourteen: Comparing bigrams and trigrams

One can compare the bigrams and trigrams of [insert your df name] to the bigrams and trigrams found in the full, original dataframe for coronavirus with over 59,000 observations. Table 1 below shows the comparison of bigrams. Table 2 below highlights the differences in trigrams. [fill in the tables below with the appropriate information]

Table 1: Bigram comparison

|  |  |
| --- | --- |
| Top 5 ‘meaningful’ bigrams for [insert your df name] | Top 5 ‘meaningful’ bigrams found in full covid dataset |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table 2: Trigram comparison

|  |  |
| --- | --- |
| Top 5 ‘meaningful’ bigrams for [insert your df name] | Top 5 ‘meaningful’ bigrams found in full covid dataset |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Stage Fifteen: Making a ngram function

One way that a researcher or analyst can save energy and make work more efficient is by creating a function. The below images demonstrate how one would create a function that always located the 5-gram within any dataset where the tweets were in a column called text. The first image is a function that is unfiltered and produces all 5-word combinations. The second image is a function that automatically filters the original dataframe as to isolate English-speaking users only.

Image 6: The unfiltered function for a 5-word ngram

[insert image of function code]

Image 7: The function for a 5-word ngram that filters for English

[insert image of function code]

Conclusion

[Write a conclusion that summarizes the overall process and your most interesting finding]

Bibliography

Borra, E., & Rieder, B. (2014, May 19). *Programmed method: Developing a toolset for capturing and analyzing tweets*. Aslib Journal of Information Management. Retrieved March 28, 2022, from https://www.emerald.com/insight/content/doi/10.1108/AJIM-09-2013-0094/full/html

Rinker, T. W. (2021). sentimentr: Calculate Text Polarity Sentiment version 2.9.0. <https://github.com/trinker/sentimentr>

Hadley Wickham, Romain François, Lionel Henry and Kirill Müller (2022). dplyr: A Grammar of Data Manipulation. R package

version 1.0.8. <https://CRAN.R-project.org/package=dplyr>

Matt Dowle and Arun Srinivasan (2021). data.table: Extension of `data.frame`. R package version 1.14.2.

<https://CRAN.R-project.org/package=data.table>

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

Taiyun Wei and Viliam Simko (2021). R package 'corrplot': Visualization of a Correlation Matrix (Version 0.92). Available

from <https://github.com/taiyun/corrplot>

Silge J, Robinson D (2016). “tidytext: Text Mining and Analysis Using Tidy Data Principles in R.” \_JOSS\_, \*1\*(3). doi:

10.21105/joss.00037 (URL: https://doi.org/10.21105/joss.00037), <URL: <http://dx.doi.org/10.21105/joss.00037>>.

Hadley Wickham and Maximilian Girlich (2022). tidyr: Tidy Messy Data. R package version 1.2.0.

<https://CRAN.R-project.org/package=tidyr>