ANALYZING DIFFERENT BIBLE TRANSLATIONS

DAVIS BUSTEED

LING 360 – FINAL PROJECT

PROJECT GOAL

I wanted to choose a final project that would be both interesting and challenging. I decided to investigate which translation of the Bible is most similar to the King James Version (KJV). I've always found it curious that there are so many different translations of the Bible, and thought that it would be interesting to see if I could use techniques such as the Levenshtein distance to systematically analyze which version of the Bible were most similar.

PROCESS

While completing this project, I wasn't very aware of the steps I was taking to accomplish my goal. Looking back, I realized that my efforts could be categorized into the following groups: research and planning, web scraping, and analysis.

RESEARCH & PLANNING

After solidifying my project idea, I make sure that it would be possible. I first looked for a corpus of different Bible translations, but was unable to find any. Knowing that I would have to build my own corpus, I started to look online for websites that let users read the Bible in different translations. I evaluated these sites for "scrape-ability," such as whether or not the content was properly tagged. I decided to use www.biblegateway.com, which fit all of these requirements.

I also made some plans about how I wanted my corpus to be structured, as well as how my Python scripts would be separated. I knew it would be unwise to create a corpus and run analysis in the same file, so I made sure to split up my logic by different functionalities. A high-level diagram of the connections and "flow" of the different scripts can be seen in Appendix A.

WEB SCRAPING

After looking at the HTML layout of the bible pages, I started work on the web scraper. The concept behind this step was simple, but was made complex due to its scale. The scraper itself was split into two sections. The first of which scraped the table of contents for each version of the Bible, storing the version name along with each book and its respective number of chapters.

The second step used this dictionary of name, books, and chapters, to build the URL necessary to grab the text for each chapter from www.biblegateway.com. After which, the HTML was cleaned so that only plaintext was left, which was then saved into a text file. This step of corpus creation took about one hour to run (see Appendix B).

ANALYSIS

After creating the corpus, I started analyzing the text. Although my primary objective was to find which translation was the most similar to the KJV based on the Levenshtein distance, I looked at

some other interesting linguistic features such as sentiment. The process of analyzing the Levenshtein distance between the different Bibles was broken down into three scripts.

The first program cycled through all the combinations of the Bible translations, so that each could be compared to all the others. The script would set the two file paths, one for each of the versions that would be compared. Then, while looping through the possible books found in the Bible, each chapter from both versions were read into the file and compared. The Levenshtein distance scores for these comparisons were saved into a dictionary, which was saved as a CSV at the end of the script.

The second script isn't shown in the flowchart in Appendix A because it was originally done in the Python REPL shell, and formalized into a script afterwards. It simply read in the CSV data from the previous script, and outputted another CSV file that contained the distance scores for the KJV Bible only. The third script read in this CSV file, and used the MatPlotLib module to create a series of summary statistics (mean, standard deviation) and charts. As seen in Appendix A, the process of analyzing sentiment followed a similar series of steps.

CHALLENGES

The first issue I faced was in the "Research & Planning" stage. I had never made a corpus before, and I didn't know how to go about it. I wanted the corpus to be easy to write, as well as easy to read in by other programs. I overcame this challenge by looking at some different corpora online to see how they were structured. I then imagined running Python scripts with this hypothetical corpus structure in mind, and realized that it would probably work very well.

I ran into my second challenge during my analysis of the corpus. I found that many books had a Levenshtein distance score below 30. When I looked into it, I saw that some "chapters" only had a line or two of text. On www.biblegateway.com, I saw that many chapters in the different translations were formatted as poetry, which required the use of different HTML classes and tags (see Appendix C for an example of these differences). To overcome this obstacle, I added a special condition into the scraper that checked if the chapter was in poetry format. For these instances, I wrote a separate series of steps to scrape and save the text to the corpus.

The last challenge I faced wasn't too disrupting in comparison to the success of the project, rather, it was more for the sake of presentation. I had some trouble using the MatPlotLib module when attempting to visualize the different Levenshtein distances, mostly because I wanted to display six different charts at once. I overcame this challenge by reading sections of the MatPlotLib documentation and looking at examples online. I wouldn't say that I am now a MatPlotLib expert, but I am a lot more comfortable with making complex visualizations.

CONCLUSION

Taking LING 360 has been a great experience. The principles and skills I learned in this class were not only extremely useful for completing this project, but I expect they will also be beneficial in my future career.

PROJECT COMPLETION

Although there were a lot of interesting analytical techniques I learned in this class, the most important subject that helped me complete my project was the lecture on Levenshtein distance. This method of comparing texts was crucial to my project completion. Learning how to install and use the FuzzyWuzzy module gave me the confidence to use it in this project.

Another principle of text processing that was useful for my project completion was the importance of splitting up functionalities into different scripts. When we were working with Twitter, we learned that it is smart to have your data collecting logic separated from your analysis. My project would've been very difficult to complete if I didn't follow this practice.

FUTURE CAREER

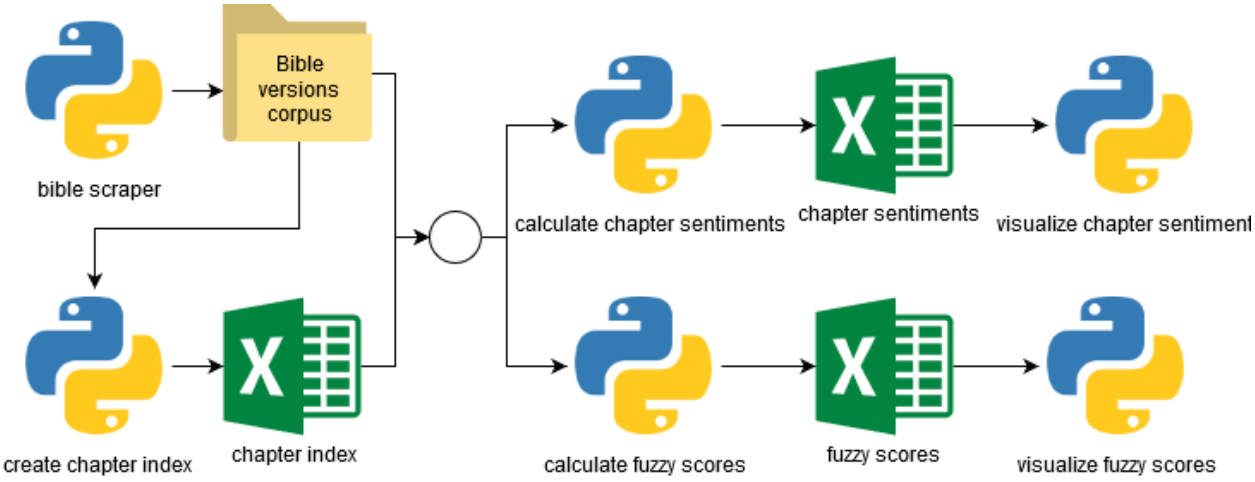
I also learned that it is important to create a project plan before starting work on the project, a practice that will definitely be useful in my future career. Even for small exercises in class, we were often to asked to think about how we would solve a certain problem for a minute or two before we started writing code.

The final lesson I learned from LING 360 is the importance of talking with others when working on a project. In our LING 360 section, sharing ideas and insights was very commonplace. I believe that applying this principle in my future career will expose me to new ideas and opportunities.

WORD COUNT: 1193 (Not including appendices & headings)

APPENDICIES

APPENDIX A – OVERALL STRUCTURE OF PROCESS



APPENDIX B – WEB SCRAPER IN ACTION

NOTE: `scraper.py` was later renamed to `bible_scaper.py`

```
C:\Users\buste\Desktop\LING_360\final_project> python scraper.py

finding number of chapters per book per version

100% [************************************************************] Elapsed Time: 0:00:02

scraping Bible text for each chapter for each book for each version
(this might take a while)

100% [************************************************************] Elapsed Time: 0:55:08
```

APPENDIX C – POETRY FORMATTING FOUND IN SOME CHAPTERS

(from left to right) normal formatting and poetry formatting for Lamentations 2

Lamentations 2 King James Version (KJV)

2 How hath the LORD covered the daughter
anger, and cast down from heaven unto the
and remembered not his footstool in the day

2 The LORD hath swallowed up all the habitat
pitied: he hath thrown down in his wrath the
daughter of Judah; he hath brought them dc
polluted the kingdom and the princes thereo

3 He hath cut off in his fierce anger all the hc
back his right hand from before the enemy,
like a flaming fire, which devoureth round ab

4 He hath bent his bow like an enemy: he stc
adversary, and slew all that were pleasant tc
the daughter of Zion: he poured out his fury

5 The LORD was as an enemy: he hath swallo
swallowed up all her palaces: he hath destrc
hath increased in the daughter of Judah mou

Lamentations 2 New International Version (NIV)

[a]How the Lord has covered Daughter Zi
with the cloud of his anger[b]!
He has hurled down the splendor of Israe
from heaven to earth;
he has not remembered his footstool
in the day of his anger.

2 Without pity the Lord has swallowed up
all the dwellings of Jacob;
in his wrath he has torn down
the strongholds of Daughter Judah.
He has brought her kingdom and its prin
down to the ground in dishonor.

3 In fierce anger he has cut off
every horn[c][d] of Israel.
He has withdrawn his right hand
at the approach of the enemy.
He has burned in Jacob like a flaming fire
that consumes everything around it.