Jeffery Dirden
L02
ITAI 2372
June 7, 2025

Reflection

This lab was my first actual deep dive into text preparation for NLP, and I came away with a better understanding of how important it is in any real world language application. Preprocessing appeared to be a simple cleanup step at first, but I soon discovered it was the foundation for making raw text valuable and understandable to computers. One key insight is that preprocessing options can significantly impact model performance. It's not just about reducing noise; it's about altering language in such a manner that it retains its meaning while also allowing the model to read it more accurately. Stop word removal, for example, minimizes clutter, but if not done correctly, it might eliminate significant context, particularly in sentiment analysis (such as removing "not" from "not good"). The contrast between NLTK and spaCy was particularly striking: NLTK is simpler and faster, but has less features, whereas spaCy provides more structured and extensive linguistic information. This demonstrated how tool selection can effect both the quality and speed of downstream NLP activities.

One of the most difficult issues was determining whether to eliminate components such as emojis, punctuation, and contractions. In other circumstances,

such as cleaning social media text, deleting emojis appeared to minimize noise, but I also noticed how it may remove sentiment clues necessary for opinion analysis. Another issue was learning how to understand and compare the results of various processing modes (minimal, standard, aggressive). Each setting had advantages and disadvantages, and I had to consider carefully how much preprocessing was "too much.” The lab made me think about all of the systems I use that rely on natural language processing, such as search engines, voice assistants, and even spam filters. I noticed how standardizing language through lemmatization or stemming may improve search performance, while deleting stop words could assist focus on only the most significant keywords. In particular, I linked the aggressive pipeline to technologies such as Google Search, where finding the basis of a term is more important than knowing every detail.

Questions that I cam up with are, when is stemming preferable to lemmatization for word reduction? Should punctuation always be deleted in activities like chatbots, or may it aid with tone detection? How do models strike a balance between fast processing and accuracy in real time applications such as customer care bots? NLTK vs spaCy: While NLTK appears to be a lighter, easier-to-use tool, spaCy is more powerful and accurate for complex tasks such as entity recognition or syntax analysis. Stemming versus Lemmatization: Stemming is faster but can result in odd terms ("fli" for "flies"), whereas lemmatization is slower but preserves valid word forms and deeper meanings ("better" to "well"). This contrast helped me grasp the difference between speed and accuracy, which is an important balance to establish depending on the task at hand. In future projects, particularly those including customer reviews or social media analysis,

I'd probably use the Standard processing pipeline with lemmatization and stop word removal. This preserves the focus on meaningful phrases while removing unnecessary clutter. I've also learned to adapt the cleaning procedure to the type of text - sophisticated cleaning for social media, lighter cleaning for official documents.

This lab showed me that there are no "one-size-fits-all" solutions to NLP preprocessing. Every decision you make, from how you tokenize to whether you eliminate emojis, has an impact on the final result. As I continue to work on NLP projects, I will become more conscious of the significance of context, task goals, and data style when creating preprocessing pipelines. I'm eager to apply these skills to larger projects and see how preprocessing might be improved for better AI understanding.