Robert Marshall
CS410 Fall 2022
rfm4@illinois.edu

Tech Review: NLTK

A Natural Language Toolkit for Python

Introduction

The world is full of written text information—books, news articles, blogs, research papers, webpages. Text information can contain a wealth of insights if analyzed with the right tools and knowledge. Natural language processing refers to the subset of data science that focuses on processing written text information.

Python has become one of the most ubiquitous programming languages in today's world, gaining popularity for its ability to make difficult tasks relatively simple to achieve through code. Python also has great support in the form of external libraries tailored for specific uses and niches; Python has over 137,000 libraries as of last month[1]. Users can easily install and use these Python libraries to achieve more targeted goals.

Natural Language Toolkit (NLTK) is among the most popular Python libraries for natural language processing tasks. NLTK is a free, open-source Python library that even comes with a free book available through NLTK's webpage to help users get started (https://www.nltk.org/book/). According to NLTK's webpage, NLTK "provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum[2]."

In this paper, we take a closer look at NLTK, studying what natural language processing tasks NLTK supports, the areas where NLTK helps users the most as well as the limitations of NLTK, the tasks for which we might want to seek other tools.

Review

NLTK has great built-in functionality for tokenization. Tokenization splits text data into smaller, simpler units such as words when preparing text data for analysis. NLTK provides simple functions within the "tokenize" module that can tokenize text data by word or by sentence. You can easily extract a list of tokenized words from text data.

NLTK also offers streamlined stemming of text data. Stemming is "a text processing task in which you reduce words to their root, which is the core part of a word[3]." The words "writing" and "writer" for instance share the root "write." The "stem" module within NLTK offers easy-to-use functions that output a list of word roots from an input list of tokenized words. NLTK's stemming algorithms do have some flaws however and don't always yield perfect results[3]. If your goals

require extremely accurate stemming, it could be worthwhile to explore other tools. NLTK also provides a lemmatizing function which "reduces words to their core meaning[3]" and can often yield more practical results compared to stemming.

NLTK provides user-friendly part-of-speech tagging, which labels each word in your data with that word's part of speech (noun, verb, adjective, preposition, …). NLTK's part-of-speech tagging seems to work rather nicely, but it can't always resolve part-of-speech ambiguity for words like "split" that can be a noun, a verb, or an adjective in different contexts.

Another important task in NLP is filtering stop words. Stop words include common words such as "the", "a", or "is" that don't have much effect on the overall meaning of a text. NLTK provides a built-in list of stop words which makes it simple for users to filter stop words from their text data to focus more on the meaningful words.

NLTK supplies users with straightforward named entity recognition (NER). NER tags certain nouns or noun phrases with the type of named entity. "Mount Everest" is a "location" named entity for example, while "President Obama" is a "person" named entity in NLTK[2].

NLTK also offers many built-in corpora, "covering everything from novels hosted by Project Gutenberg to inaugural speeches by presidents of the United States[3]." These corpora serve as great test datasets for learning about NLP, making NLTK a popular library in education and research settings[4].

NLTK provides a "sentiment" module with helpful tools for sentiment analysis. NLTK offers a few pre-trained sentiment analyzers but also makes it fairly straightforward for users to customize sentiment analysis[5].

References

1. Great Learning Team. "Top 30 Python Libraries to Know in 2023." Great Learning Blog: Free Resources What Matters to Shape Your Career!, 31 Oct. 2022, www.mygreatlearning.com/blog/open-source-python-libraries.
2. Steven Bird, Ewan Klein, and Edward Loper (2009). Natural Language Processing with Python. O'Reilly Media Inc. https://www.nltk.org/book/
3. Real Python. (2022, February 18). Natural language processing with python's NLTK package. Real Python. https://realpython.com/nltk-nlp-python/
4. Tomasz Bak. (2019, October 14). Python NLP libraries: Features, use cases, pros and cons. Medium. Retrieved November 6, 2022, from https://medium.com/@tomaszbak/python-nlp-libraries-features-use-cases-pros-and-cons-da36a0cc6adb
5. Real Python. (2022, September 1). Sentiment analysis: First steps with Python's NLTK library. Real Python. Retrieved November 6, 2022, from https://realpython.com/python-nltk-sentiment-analysis/