CS410: Tech Review

Topic: Comparison of NLP Toolkits

Soumya Kanti Dutta (skdutta2@illinois.edu)

What is NLP (Natural Language Processing)

Natural languages are built to communicate among human beings. Human can comprehend the natural Languages because human has common sense to understand the meaning of words,sentences and word phases. But machine or computer does not have the common sense or large historical databases to infer Knowledge to comprehend the word relationships and meaning of the sentences or phases.

Natural Language Processing (NLP) makes it possible for computers to understand the human language(eg. English or Bengali).

Probably, the most popular examples of NLP in action are virtual assistants, like Google Assist, Siri, and Alexa. NLP understands and translates the human language, like *"Hey Siri, where is the nearest gas station?"* into numbers, making it easy for machines to understand.

Now the question arises how these tools or systems are able to comprehend complex human natural Language to machine comprehensive languages.

Natural Language Processing (NLP) applies two techniques to help computers understand text: **syntactic analysis and semantic analysis**. We are not discussing in detail of these techniques here. Because this is out of scope of this topic.

Although there are many techniques are available "bag of words" representation or the unigram language model is the most effective way to achieve this goal for sentiment analysis or test mining (text clustering or text/topic categorization). After applying these NLP tools to preprocess the text data, we can use count vectorization (Term Frequency) or TF-IDF (Term Frequency and Inverse Document Frequency) heuristics to transform the text data to machine compatible format.

Syntactic Analysis

Some of its main sub-tasks include:

- **Tokenization** consists of breaking up a text into smaller parts called *tokens* (which can be sentences or words) to make text easier to handle.
- **Part of speech tagging (PoS tagging)** labels tokens as *verb, adverb, adjective, noun*, etc. This helps infer the meaning of a word (for example, the word "book" means different things if used as a verb or a noun).
- **Lemmatization & stemming** consist of reducing inflected words to their base form to make them easier to analyze.
- **Stop-word removal** removes frequently occuring words that don't add any semantic value, such as *I, they, have, like, yours*, etc.These words occuer everywhere but do not entitle any value.

Semantic Analysis

- **Word sense disambiguation** tries to identify in which sense a word is being used in a given context.

each other in a text.

Popular NLP Tools to handle this task:

- **NLTK(Natural Language Toolkit):** The Natural Language Toolkit (NLTK) with Python is one of the leading tools in NLP model building. Focused on research and education in the NLP field. We can perform tokenization,POS tagging, stemming, stop word removal techniques during data preprocessing task.

Advantages of NLTK:

- Most well known python NLP library used for text preprocessing
- Many third party extensions
- Fast sentence tokenizations
- Support large number of natural languages

Disadvantages of NLTK:

- Quite slow
- Does not provide neural network support. We can't use NLTK library for heavy duty NLP taks like language translation, Text summarisations etc. For language translations generally we use BERT and attention model (with modifications of LSTM and GNN)
- No integrated word vectors or word embeddings which are often used by wordtoVec or Stanford Glove word embediing techniques

- **Spacy:** One of the newest open-source Natural Language Processing with Python libraries on our list is SpaCy. It's lightning-fast, easy to use, well-documented, and designed to support large volumes of data, not to mention, boasts a series of pretrained NLP models that make your job even easier. Support available for python and cython. As Spacy is written is cython, it's processing speed is much higher.

Advantages of Spacy:

- Processing is fast
- Very good documentation available and easy to comprehend and implement.
- Prebuild model is readily available . You just need to load the model. Eg. For English there are three pre built models are available (small, medium and large)
- Uses Neural networks for training some models
- Great community support
- Spacy also have some framework, ie. you can use regular expressions to identify some word phases.

Disadvantages of Spacy:

- Sentence tokenlization is slower than NLTK
- Does not support many languages. Currently it has only support for 7 languages.
- Lack of flexibility compared to NLTK.

- **GenSim:** Gensim is a highly specialized Python library that largely deals with topic modeling tasks using algorithms like Latent Dirichlet Allocation (LDA). It's also excellent at recognizing text similarities, indexing texts, and navigating different documents.

Advantages of GenSim:

- Works with large datasets and
- Provide TF-IDF vectorisations and word2vec,latest semantic analysis
- Support deep learning

Disadvantages of GenSim:

- Mainly designed for unsupervised text modeling.
- Does not support most of NLP text preprocessing tasks
- Lack of flexibility compared to NLTK and spacy in terms of text preprocessing task

Conclusion:

In this topic we are trying to illustrate the advantages and disadvantages of some NLP toolkits which are readily available in market and have been grown and evolved over the past few years. We need to choose The NLP library or toolkit based on the requirement and purpose because these toolkits are also built to solve a particular problem. As a result, we need to identify the problem statement and should be very clear what we are going to achieve. Based on the requirement we can choose particular NL toolkit.

In real life, we may need to apply the hybrid approach as well. Eg. Some problems should be addressed by NLK and some are addressed by Spacy or GenSIm or other popular NLP toolkit techniques.

I have also faced some problem while working on a text mining project. Let's take an example.

NLTK, Spacy,GenSim is working fine for a general news article written in English. If you need to build a topic mining model from a branch domain specific documents (even though this documents are written in English) no toolkit will address the problem. I used spacy to tokenize the document sentences and after that I need to create a domain specific language model so that these key words get precedence even though these tokens do not signify much importance in English literature.

In a nutshell, choosing a NLP library is completely dependent on the problem you want to solve/address.