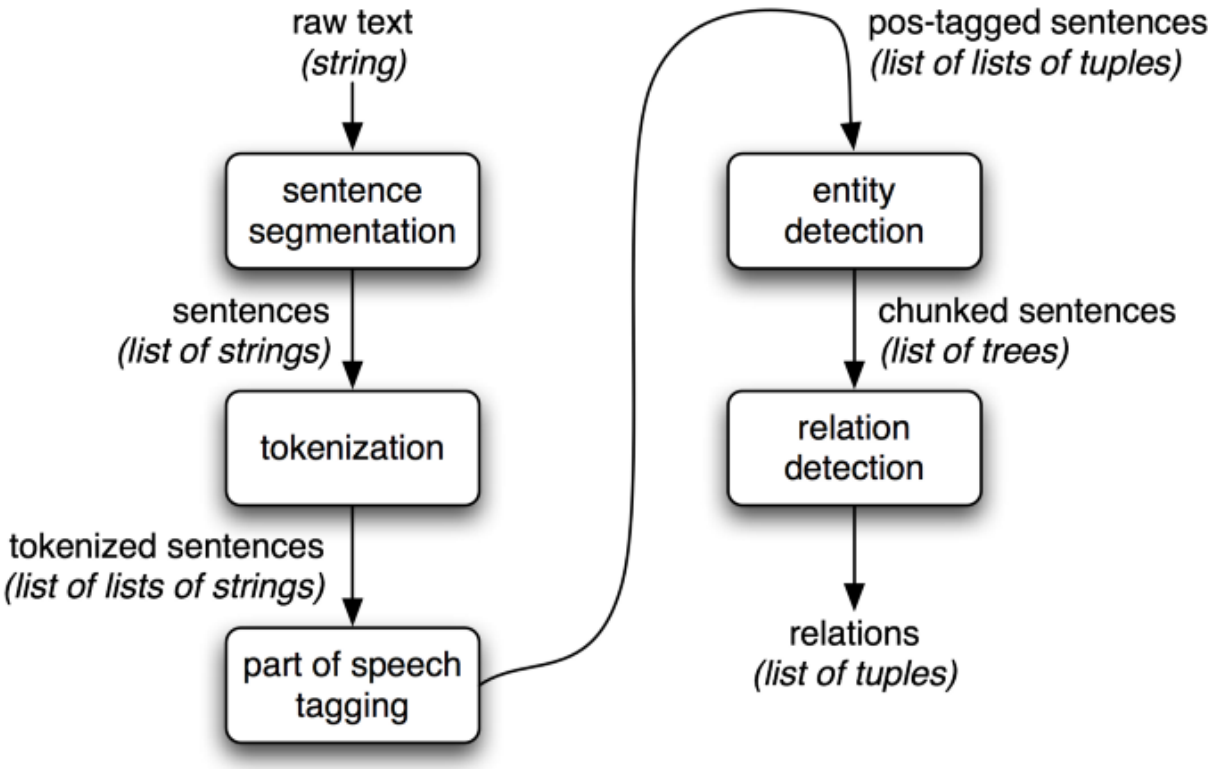
NLTK for Information Extraction

The natural language toolkit (NTLK) has been recognized as one of the leading platforms for developing Python programs that have the ability to work collaboratively with human language information. NTLK presents users with non-complex interfaces that comprise of dozens of lexical and corpora sources, which include semantic reasoning, parsing, stemming, tagging, and discussion forums among others. The amount of natural language information in digital servers is staggering and is increasing constantly as time passes. Retrieving such data is quite cumbersome because of the unstructured nature of the data (NP para 1). A tool such as NTLK enables extraction of structured data from unstructured datasets.

Information is stored in different shapes and sizes which determines the ease in extraction. Natural language data such as paragraphs of text is usually unstructured, which makes it difficult to retrieve without specialized tools such as NTLK. Due to the complexity of extracting structured human language data, it is necessary to convert it into structured format to aid in systematic retrieval. Data retrieval thus entails splitting the raw natural language sentences with the help of a sentence segmenter, and further using a tokenizer for subdividing it into words. The sentence then undergoes the tagging process where each part is tagged to speech tags. The next process involves entity detection where interesting entities in a sentence are identified. The last step involves searching for different relations among entities in a process called relation detection (Jeffy para 6). The figure below outlines architecture for the unstructured data information extraction process.



Source: Jeffy

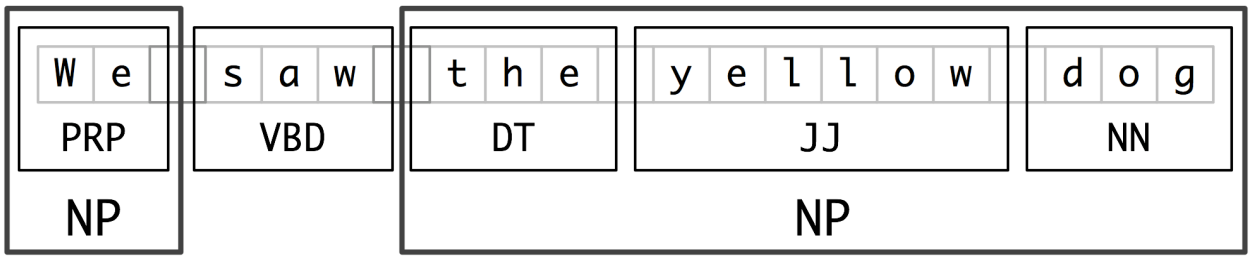
In performing the first three processes, a simple illustrational function was developed that shows how default NTLK connects segmenter, tokenizer, and part of speech tagging as shown in the extract below.

```
>>> def ie_preprocess(document):
...     sentences = nltk.sent_tokenize(document) ❶
...     sentences = [nltk.word_tokenize(sent) for sent in sentences] ❷
...     sentences = [nltk.pos_tag(sent) for sent in sentences] ❸
```

Next, in entity detection, entities that show relations with others are segmented and labeled. These enable the identification of definite noun phrases, however, in some tasks indefinite nouns are also considered.

Chunking is the basic technique that is used for detection of entities and involves segmenting and labeling of multi-token sequences as show in the in the diagram below. The

small boxes indicate tokenization on the world level and part-of-speech tagging (13). On the other hand, the larger boxes show chunking on the higher-level and are called chinks.



Another important process of in NLTK data extraction is the noun phrase chunking, which follows a rule that establishes whether context that it assumes signify a noun phrase. The chunk is tagged as noun phrase is the function identifies “determiner” that if followed by an “adjective”. The part-of-speech tags are most crucial in noun phrase chunking. With noun phrase chunking, it is necessary to take into consideration the chunk grammar as it provides the rules that determine how sentences ought to be chunked.

Chinking is another process used by NLTK in extraction of data. Sometimes during data extraction, it is easier to identify what needs to be excluded from a data chunk other than selecting the parts are required. A chink is therefore a string of tokens that are excluded from a chunk.

NLTK has been identified as an important application linguistics, researchers, students, and engineers among other users. The accessibility of the application in various operating systems such as Mac OS X, Linux, and Windows makes it available to a wide array of users. Further, NLTK with Python serves as a practical approach to introduce novices in language processing programming.

Works Cited

Jeffy, Sam. Chunking and Extracting information using NLTK — PART -6. 1 June 2020, https://medium.com/@jeffysam02/chunking-and-extracting-information-using-nltk-part-6-5ecceeb4aac4. Accessed on 10 Nov. 2020.

NP. NLTK 3.5 documentation: Natural Language Toolkit. 13 April 2020. https://www.nltk.org/. Accessed on 10 Nov. 2020.