**Technology review document describes different libraries used in this software.**

**NLTK**

The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language. NLTK is a leading platform for building Python programs to work with human language data. NLTK is a very big library holding 1.5GB and has been trained on a huge data. It has over 50 corpora and lexicons, 9 stemmers, and dozens of algorithms to choose from. It can provide you with different dataset in multiple languages which you can deploy according to the functionality you require.

Natural language processing (NLP) is about developing applications and services that are able to understand human languages. We are talking here about practical examples of natural language processing (NLP) like speech recognition, speech translation, understanding complete sentences, understanding synonyms of matching words, and writing complete grammatically correct sentences and paragraphs.

As all of we know, millions of gigabytes every day are generated by blogs, social websites, and web pages. There are many companies gathering all of this data to better understand users and their passions and make appropriate changes. These data could show that the people of Brazil are happy with product A, while the people of the US are happier with product B. With NLP, this knowledge can be found instantly (i.e. a real-time result). For example, search engines are a type of NLP that give the appropriate results to the right people at the right time.

So there are many such applications where NLP is playing very crucial. Which indicates need of one strong, fast, efficient library which can help to process and understand the data. NLP is one of the most widely used open source library to achieve this.

There are many such libraries like GATE, LingPipe, Stanford core nlp, OpenNLP and so on. I have used all of these library for one or another task in past. Frankly saying, there is no one library apart from NLTk which has so many features and one solution to most of all NLP text processing requirement.

Recently Spacy has also shown good interest for NLP. But NLTK is proven and research/industry preferred toolkit for NLP operation.

NLTK is a string processing library, it takes strings as input and returns string as output. Whereas, spaCy uses object-oriented approach. As output spaCy returns document object with word and sentence objects. NLTK usually provides a bunch of algorithms for one particular problem, whereas spaCy keeps the best algorithm for a problem

Let's talk about how to install, how to use and what can we do using NLTK.

**1. Install NLTK**

If you are using Windows or Linux or Mac, you can install NLTK using pip: # pip install nltk.

You can use NLTK on Python 2.7, 3.4, and 3.5 at the time of writing this post. Alternatively, you can install it from source from this tar.

To check if NLTK has installed correctly, you can open your Python terminal and type the following: Import nltk. If everything goes fine, that means you've successfully installed NLTK library.

Once you've installed NLTK, you should install the NLTK packages by running the following code:

*import nltk*

*nltk.download()*

This will show the NLTK downloader to choose what packages need to be installed.

**Example - How to use it**

Let’s see one example how to use nltk for sentence tokenization.

from nltk.tokenize import sent\_tokenize

mytext = "Hello Adam, how are you? I hope everything is going well. Today is a good day, see you dude."

print(sent\_tokenize(mytext))

Output

['Hello Adam, how are you?', 'I hope everything is going well.', 'Today is a good day, see you dude.']

Likewise there are many features. Here is the list

* Tokenize Text Using NLTK
* Tokenize Non-English Languages Text
* Get Synonyms, Antonyms From WordNet
* NLTK Word Stemming
* Stemming Non-English Words
* Lemmatizing Words Using WordNet
* Stop word removal
* Part of Speech taggin
* NER Extraction
* Parse Tree Creation
* Sentiment Analysis

**Scikit learn**

**What is Scikit learn**

scikit-learn is a free-to-use machine learning module for Python built on SciPy. It is a straightforward and effective tool for data mining and data analysis. Because it is released with a BSD license, it can be used for both personal and commercial reasons.

With scikit-learn, users are able to conduct a variety of tasks under different categories like model selection, clustering, preprocessing, and more. The module provides the means to complete implementations.

Moreover, scikit-learn has an extensive use. It is being utilized by big companies in different industries like music streaming, hotel bookings, and more. This means that users can integrate algorithms in the platform to their own applications.

**Some benefits of Scikit learn**

**Free Platform**

Because scikit-learn is released with a BSD license, it can be used for free by everyone. This license has minimal restrictions; therefore, users can utilize it to design their applications and platforms with little worry over limitations.

**Industrial Use**

scikit-learn is a helpful platform that can predict consumer behavior, identify abusive actions in the cloud, create neuroimages, and more. It is being used extensively by commercial and research organizations around the world, a testament to its ease of use and overall advantage.

**Collaborative Library**

scikit-learn began as a one-man mission but now it is being built by numerous authors from INRIA spearheaded by Fabian Pedregosa and individual contributors who are not attached to teams or organizations. This makes the module a well-updated one, releasing updates several times a year. Users can also look forward to assistance from an international community, in case they have queries or if they hit snags in development using the module.

**Ease of Use**

Commercial entities and research organizations alike have employed scikit-learn in their processes. They all agree that the module is easy-to-use, thereby allowing them to perform a multitude of processes with nary a problem.

**API Documentation**

scikit-learn ensures that users old and new alike get the assistance they need in integrating the machine learning module into their own platforms. That is why a documentation detailing the use of its API exists that users can access anytime on the website. This makes certain developers can implement machine learning algorithms offered by the tool seamlessly.