Part 2: Step by Step Guide to NLP – Knowledge Required to Learn NLP

Chirag Goyal — June 15, 2021

Advanced NLP Project Python Text Unstructured Data

This article was published as a part of the Data Science Blogathon

Introduction

This article is part of an ongoing blog series on Natural Language Processing (NLP). In part-1 of this blog series, we complete the basic concepts of NLP. Now, in continuation of that part, in this article, we will cover some of the new concepts.

In this article, we will understand the knowledge required and levels of NLP in a detailed manner. In the last of this article, we will discuss the libraries used for NLP with the step-by-step procedure of Installation. So, if you are following this blog series from start then download that library and stay tuned with us. From the next part of this series, we will regularly use that library for implementation purposes.

This is part-2 of the blog series on the Step by Step Guide to Natural Language Processing.

Table of Contents

1. Knowledge required in NLP

- Phonetic and Phonological knowledge
- Morphological Knowledge
- Syntactic Knowledge
- Semantic knowledge
- Pragmatic Knowledge
- Discourse Knowledge
- Word knowledge

2. Levels of Natural Language Processing

- Morphological Analysis
- Lexical Analysis
- Syntactic Analysis
- Semantic Analysis
- Discourse Integration
- Pragmatic Analysis

3. Easy to use NLP Libraries

- NLTK (Natural Language Toolkit)
- spaCy
- TextBlob
- Gensim

4. Natural Language Toolkit (NLTK) Installation

1. Phonetics is the study of language at the level of sounds while phonology is the study of the combination of sounds into organized units of speech.

2. Phonetic and Phonological knowledge is essential for speech-based systems as they deal with how words are related to the sounds that realize them.

Morphological Knowledge

1. Morphology concerns word-formation.

2. It is a study of the patterns of formation of words by the combination of sounds into minimal distinctive units of meaning called morphemes.

3. Morphological Knowledge concerns how words are constructed from morphemes.

Syntactic Knowledge

1. The syntax is the level at which we study how words combine to form phrases, phrases combine to form clauses and clauses join to make sentences.

2. The syntactic analysis concerns sentence formation.

3. It deals with how words can be put together to form correct sentences.

Semantic Knowledge

1. It concerns the meaning of the words and sentences.

2. Defining the meaning of a sentence is very difficult due to the ambiguities involved.

Pragmatic Knowledge

1. Pragmatics is the extension of the meanings or semantics.

2. Pragmatics deals with the contextual aspects of meaning in particular situations.

3. It concerns how sentences are used in different situations.

Discourse Knowledge

1. Discourse concerns connected sentences. It includes the study of chunks of language which are bigger than a single sentence.

2. Discourse language concerns inter-sentential links that is how the immediately preceding sentences affect the interpretation of the next sentence.

3. Discourse language is important for interpreting pronouns and temporal aspects of the information conveyed.

Word Knowledge

1. Word knowledge is nothing but everyday knowledge that all the speakers share about the world.

2. It includes the general knowledge about the structure of the world and what each language user must know about the other user's beliefs and goals.

3. This is essential to make the language understanding much better.

In this section, we will see all the typical steps involved while performing NLP tasks. We should remember that the below section describes some standard workflow, however, it may differ drastically as we do real-life implementations based on our problem statement or requirements.

As we discussed, the source of Natural Language could be either speech (sound) or Text.

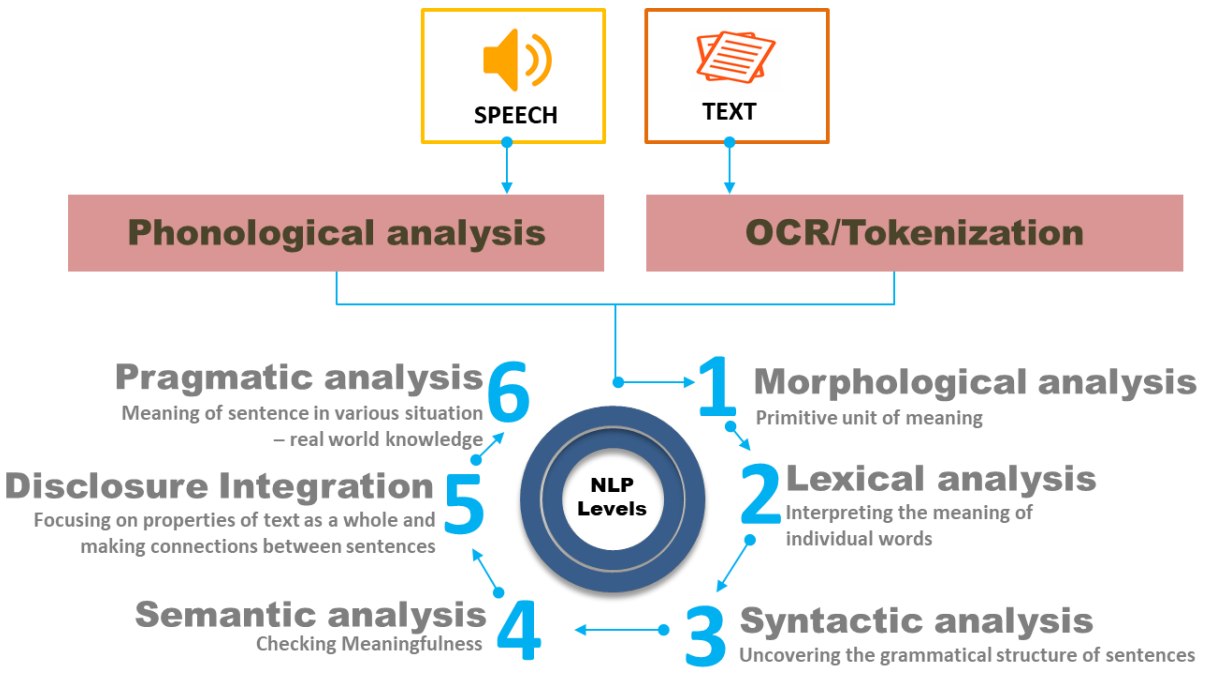


Image Source: Google Images

Phonological Analysis

This level is applicable only if the text is generated from the speech and deals with the interpretation of speech sounds within and across different words. The idea behind this step is that sometimes speech sound might give an idea about the meaning of a word or a sentence.

Therefore, it is the study of organizing sound systematically. This analysis can require a broad discussion but for you, that is out of the scope and we will not cover that portion in this blog series.

Morphological Analysis

In this analysis, we try to understand distinct words according to their **morphemes**, which are defined as the smallest units of meaning.

For Example, Consider the word: “unhappiness ”

We can be broken down into three morphemes named prefix, stem, and suffix, with each conveying some form of meaning:

The prefix **un-** refers to “not being”,

The suffix **-ness** refers to “a state of being”.

The stem **happy** is considered a free morpheme since it is a “word” on its own.

prefixes and suffixes are Bound morphemes and they require a free morpheme to which it can be attached, and can therefore not appear as a “word” on their own.

Lexical Analysis

This analysis involves identifying and analyzing the structure of words.

Let’s first understand what does Lexicon means?

To work with lexical analysis, mostly we need to perform **Lexicon Normalization**. The most common lexicon normalization practices are **Stemming** and **Lemmatization** which we will cover later in this blog series.

Syntactic Analysis

In this analysis, we will analyze the words of a sentence so as to uncover the grammatical structure of the sentence.

For Example, Consider the phrase "Colourless red idea." This would be rejected by the Syntactic analysis as the colorless word here with red doesn't make any sense.

Syntactical parsing includes the analysis of words in a particular sentence for grammar and their arrangement in a manner that shows the relationships among the words. **Dependency Grammar** and **Part of Speech tags** are the important attributes of text syntactic.

Semantic Analysis

In this analysis, we try to determine the possible meanings of a sentence based on the interactions among word-level meanings in the sentence. Some people may think it's the level which determines the meaning, but actually, all the level do.

For Example, The semantic analysis disregards sentences such as "hot ice cream".

Discourse Integration

In this analysis, our main focus is on the properties of the text as a whole that convey meaning by making connections between the different components of the sentences. It means a sense of the context. The meaning of any single sentence depends upon that sentence. It also takes the meaning of the following sentence into consideration while analyzing.

For Example, Consider the sentence "He wanted that"

Here, the word "that" in the sentence depends upon the prior discourse context.

Pragmatic Analysis

In this analysis, we explain how extra meaning is read into texts without actually being encoded in them. This analysis requires to have an idea of,

- World knowledge,
- Understanding of intentions, plans, and goals, etc.

Now, consider the following two sentences:

- The city army refused the demonstrators a permit because they feared violence.
- The city army refused the demonstrators a permit because they advocated revolution.

In the above two sentences, the meaning of "they" in both sentences is different. So, to figure out the difference, we have to utilize the world knowledge in knowledge bases and inference modules.

Therefore, Pragmatic analysis helps users to discover this intended effect (understand with the help of the above example) by applying a set of rules that characterize cooperative dialogues.

For Example, Consider the sentence "close the window?"

This sentence should be interpreted as a request instead of an order.

Easy to Use NLP Libraries

NLTK (Natural Language Toolkit)

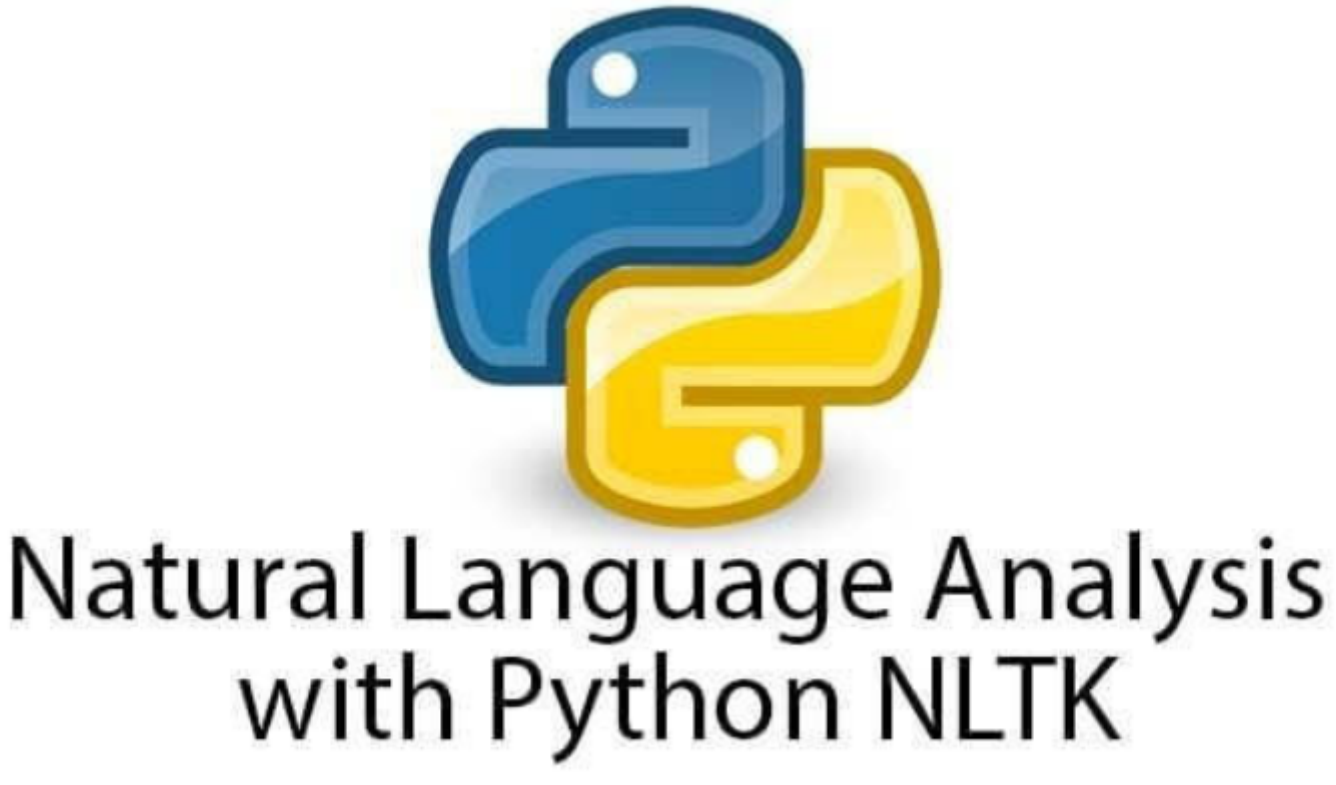


Image Source: Google Images

Generally, this Python framework is used as an education and research tool. It's not usually used on production applications.

However, this library can be used to build some exciting programs due to its ease of use.

Functionalities

- Tokenization.
- Part Of Speech tagging (POS).
- Named Entity Recognition (NER).
- Classification.
- Sentiment analysis.
- Packages of chatbots.

Applications

- Recommendation systems.
- Sentiment analysis.
- Building chatbots.

For more information, check official documentation: Link

spaCy

Image Source: Google Images

spaCy, which is an open-source natural language processing Python library. It designed to be fast nature and production-ready.

It focuses on providing software for production usage

- Tokenization.
- Part Of Speech tagging (POS).
- Named Entity Recognition (NER).
- Classification.
- Sentiment analysis.
- Dependency parsing.
- Word vectors.

Applications

- Autocomplete and autocorrect.
- Analyzing reviews.
- Summarization.

For more information, check official documentation: Link.

Gensim

Image Source: Google Images

Gensim, an NLP Python framework is generally used in topic modeling and similarity detection. It is not a general-purpose NLP library, but it handles tasks assigned to it very well.

Functionalities

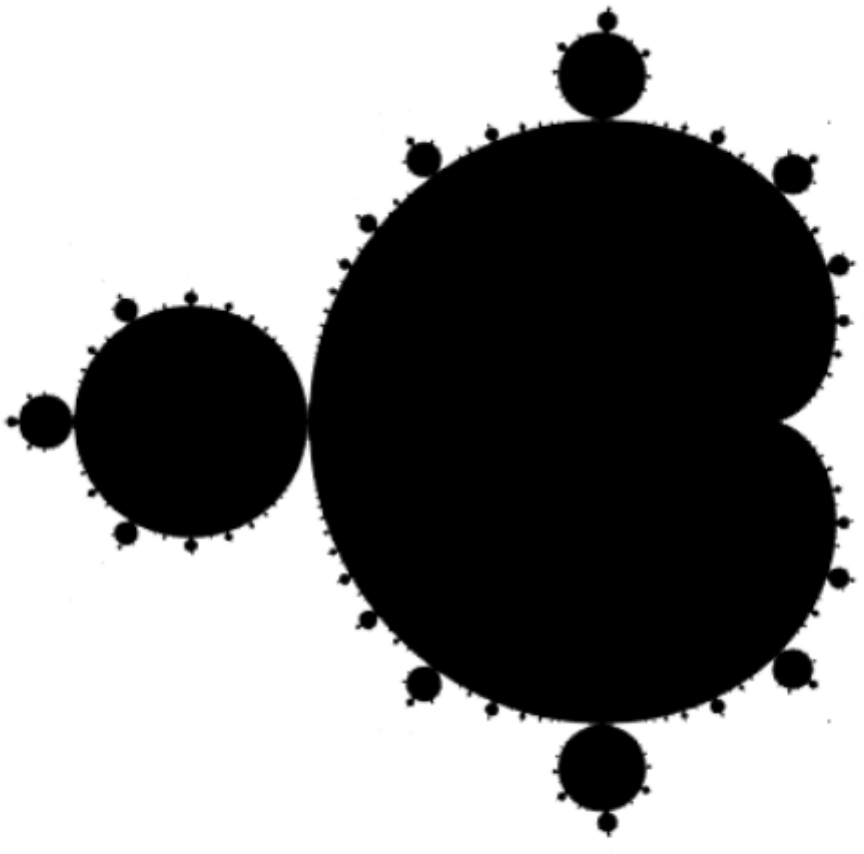
- Latent semantic analysis.
- Non-negative matrix factorization.
- TF-IDF.

Applications

- Converting documents to vectors.
- Finding text similarity.
- Text summarization.

For more information, check official documentation: Link.

TextBlob



TextBlob

Image Source: Google Images

TextBlob is a Python library that is mainly designed for processing textual data.

Functionalities

- Part-of-Speech tagging.
- Noun phrase extraction.
- Sentiment analysis.
- Classification.
- Language translation.
- Parsing.
- Wordnet integration.

Applications

- Sentiment Analysis.
- Spelling Correction.
- Translation and Language Detection.

For more information, check official documentation: Link.

Important Note about NLP Libraries:

Here we discuss only some of the libraries used for NLP tasks, but if you are interested to learn more libraries, then refer to the link .

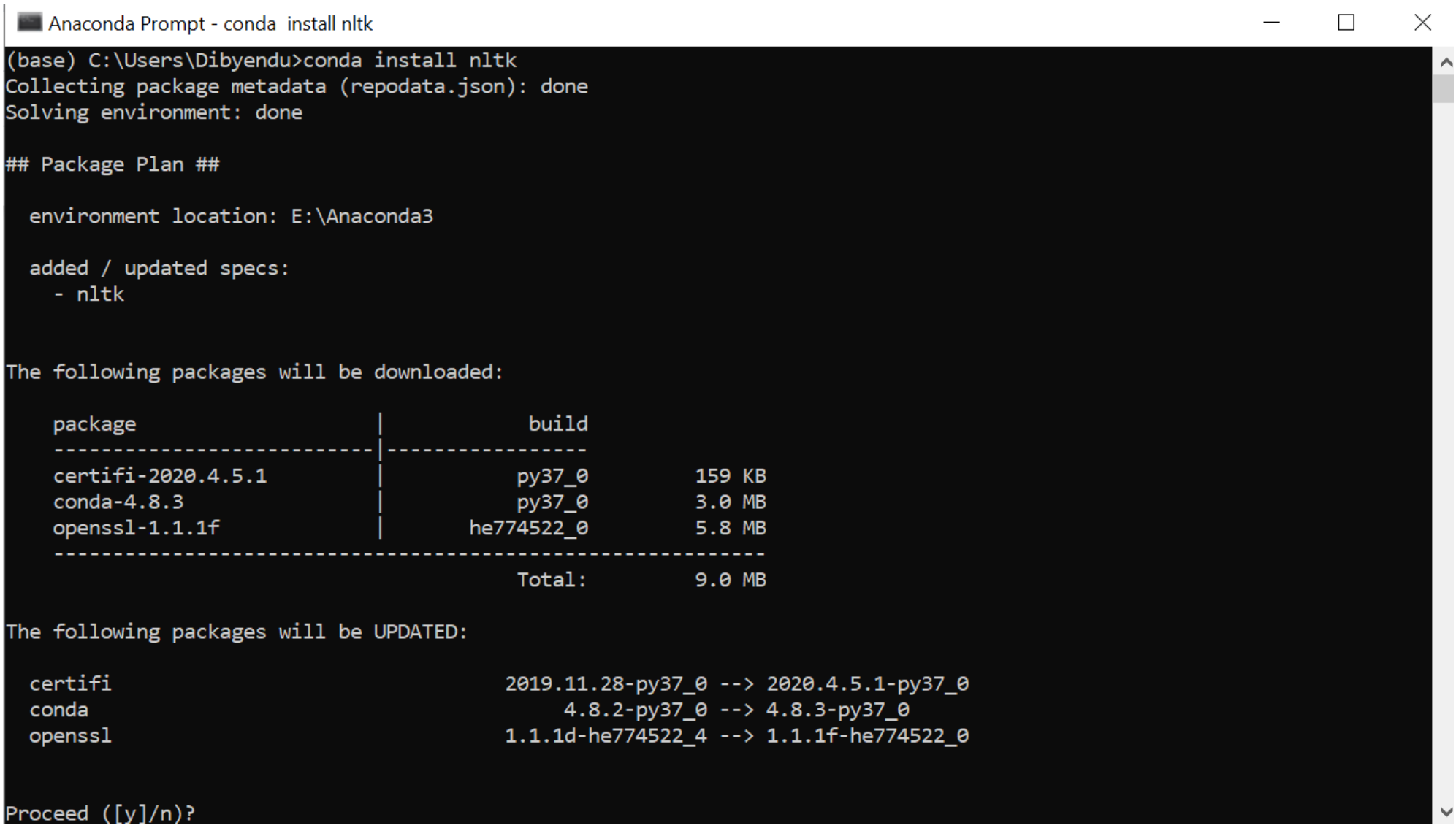
In this series, we are going to focus more on the NLTK library. So, Let's see the installation procedure of the NLTK Library for NLP.

NLTK Installation

If you are using Windows or Linux or Mac operating system, you can install NLTK using pip:

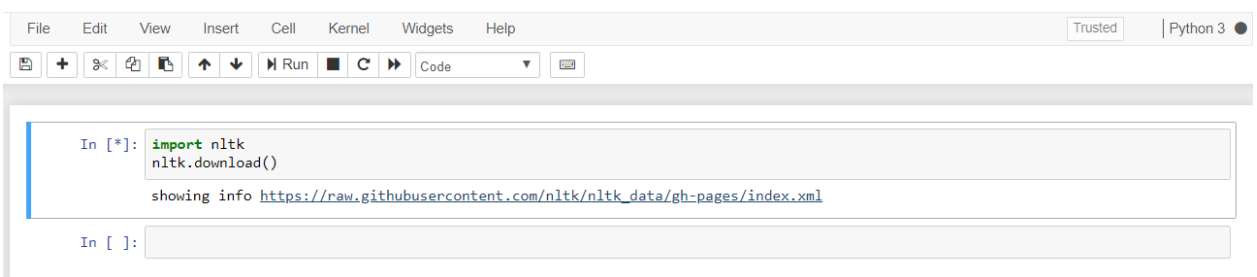
Optionally, if you use the Anaconda prompt, then try the following command:

```
$ conda install nltk
```



If everything goes in the right direction, that implies you've successfully installed the NLTK library. Once you've complete the installation process of NLTK, after that you should install the NLTK packages by running the following code:

Open your Jupyter Notebook and try to run the following command.



If same get the same output. then now it will show the NLTK downloader to choose which packages you need to be installed. You can install all packages since they have small sizes, so there is no problem at all. Now let's start the downloading.

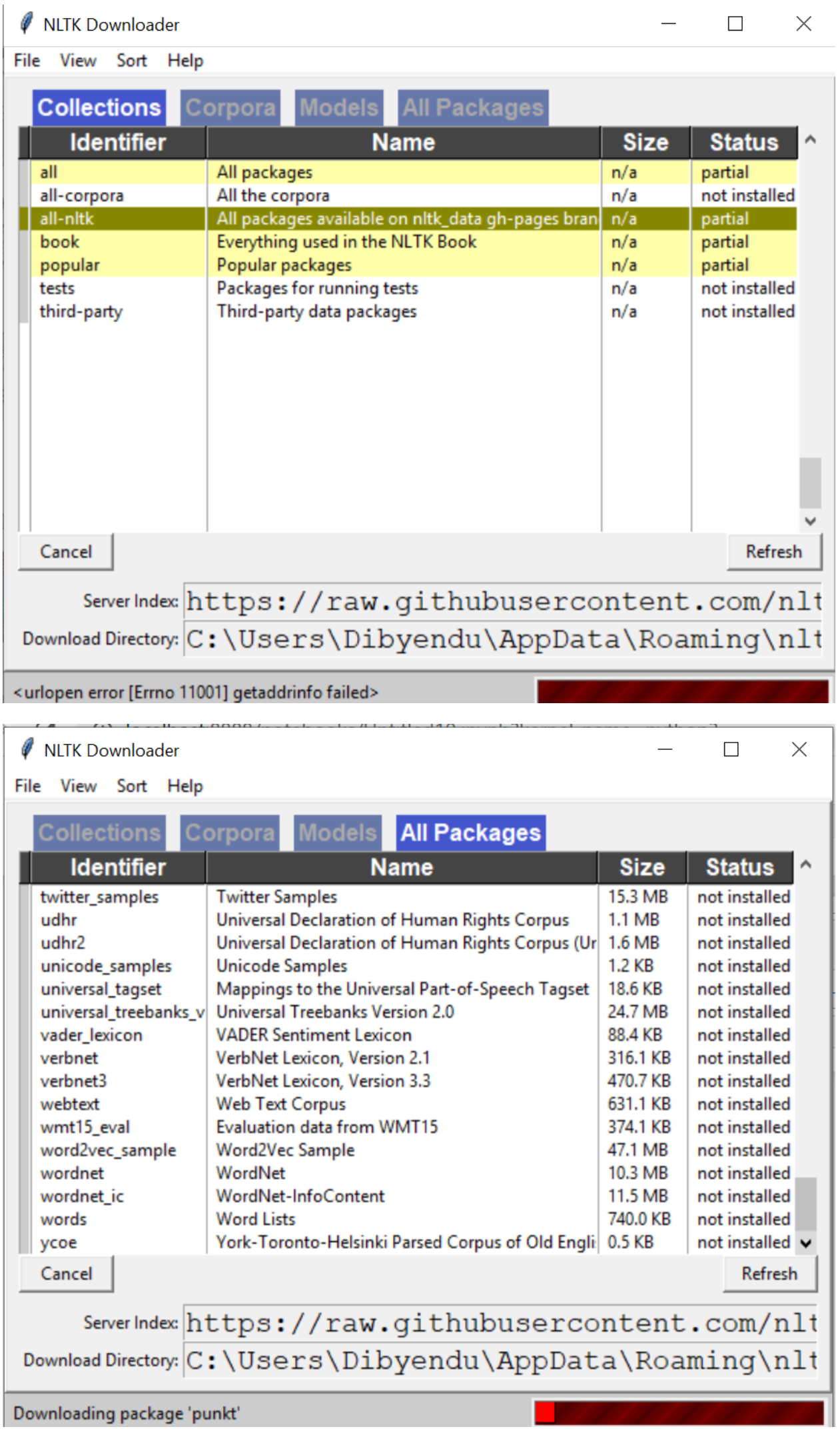


Image Source: Google Images

This completes our downloading of NLTK Library with their packages. Now, from the next articles, we will be using NLTK Library and implement different techniques involved in NLP tasks.

This ends our Part-2 of the Blog Series on Natural Language Processing!

End Notes

Thanks for reading!

If you liked this and want to know more, go visit my other articles on Data Science and Machine Learning by clicking on the **Link**