**Project Requirement and Specification**

on

**Wordnet Processing through NLP**

**(CSE V Semester Mini project)**

**2021-2022**

****

**Name: Aytijha Chakraborty**

**University Roll No: 1914009**

**Section: K**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GRAPHIC ERA HILL UNVERSITY, DEHRADUN**

**ACKNOWLEDGMENT**

I would like to thank my Project Guide Dr. Rakesh Patra Sir for his patience, support, and encouragement throughout the completion of this project and for having faith in me.

**Aytijha Chakraborty**

**Roll No.- 1914009**

**CSE-K-V-Sem**

**Session: 2021-2022**

**GEHU, Dehradun**

**ABSTRACT**

Natural Language Processing, or NLP for short, is broadly defined as the automatic manipulation of natural language, like speech and text, by software.

The study of natural language processing has been around for more than 50 years and grew out of the field of linguistics with the rise of computers.

Linguistics is the scientific study of language, including its grammar, semantics, and phonetics.

Classical linguistics involved devising and evaluating rules of language. Great progress was made on formal methods for syntax and semantics, but for the most part, the interesting problems in natural language understanding resist clean mathematical formalisms.

Broadly, a linguist is anyone who studies language, but perhaps more colloquially, a self-defining linguist may be more focused on being out in the field.

Mathematics is the tool of science. Mathematicians working on natural language may refer to their study as mathematical linguistics, focusing exclusively on the use of discrete mathematical formalisms and theory for natural language (e.g. formal languages and automata theory).

Computational linguistics is the modern study of linguistics using the tools of computer science. Computational linguistics also became known by the name of natural language process, or NLP, to reflect the more engineer-based or empirical approach of the statistical methods.

The statistical dominance of the field also often leads to NLP being described as Statistical Natural Language Processing, perhaps to distance it from the classical computational linguistics methods.

**PROJECT INTRODUCTION AND MOTIVATION**

**What is Wordnet?**

WordNet is a Corpus, a large lexical database of English Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. WordNet is freely and publicly available for download. Its structure makes it a useful tool for computational linguistics and Natural Language Processing.

**About Project**

This project (named Caramel Popcorn) is a small-scale implementation of a Movie Analysis System. This means that upon being provided the details of a movie, the probability of it making a huge success can be accurately predicted.

This project attempts to predict how successful a movie will be in the Box Office, even before it is actually released.

The pipeline of the project includes three components as of now:

1. Data Scraping [Modules Used: BeautifulSoup]
2. Feature Exploration and Engineering [Languages Used: Python]
3. NLP Classification [Modules Used: NLTK, MultinomialNB, SVC]

[Corpus Used: WordNet]

**METHODOLOGY**

As mentioned above, the model expects the Name and the complete Description of the movie to be analyzed, and classifies the movie into one of over 200 specific genres.

The data used to train this model was scraped from IMDB’s list of Top 1000 movies. This was done by parsing each webpage into an HTML document, and then extracting the useful bits of information using BeautifulSoup, a library offered by Python. The extracted data is then formatted into a Dataframe and then exported into a CSV file for further uses.

Once the required data is available in the form of a CSV file/Dataframe, it is then cleaned for our model to work with. This cleaning process is conducted in a 2-step process:

* First, any redundant/garbage column is dropped
* Next, all the Null values, Garbage values and Outliers are handled

The cleaned data is then separated into two parts, one for our model to work with, and the other being unrelated with this phase of the project.

Once we have the data required for our model, we need to preprocess it and extract the features before we move onto the next step, i.e. NLP Classification.

The data preprocessing is conducted in a few steps. First we obtain Clean\_text from the provided textual columns. To do that:

* First, we add the contents of all the Textual Columns into one Column
* Next, for each row, we turn the whole string into lower case, strip any leading/trailing whitespaces, and substitute all Punctuation marks and Escape Sequence characters with a whitespace
* Next, we look for Stopwords, i.e. words that do not add much value, and remove them from the text
* Lastly, we Lemmatize all the words, i.e. change variant-forms into their root-words

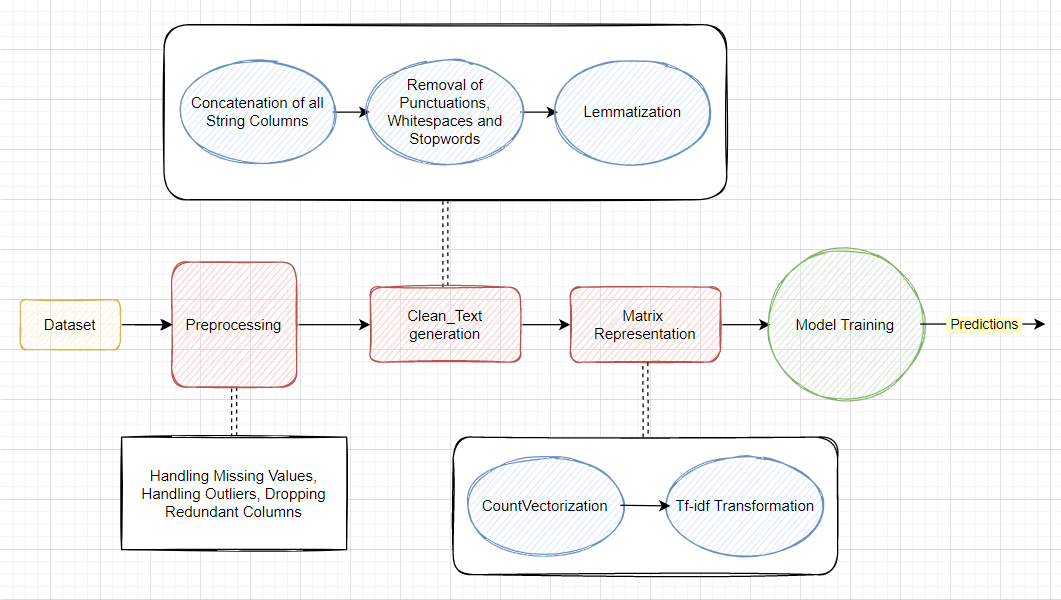
Now we have obtained a Clean\_text from the provided textual columns. Next, we need to transform this string data into a numerical format. To do that:

* First, we use CountVectorizer() to convert the collection of string data into token counts
* And then, we TfidfTransformer() to convert this count matrix into a tf-idf (term-frequency times inverse document-frequency) representation.

This is done to scale down the impact of words that occur very frequently in the string collection/document.

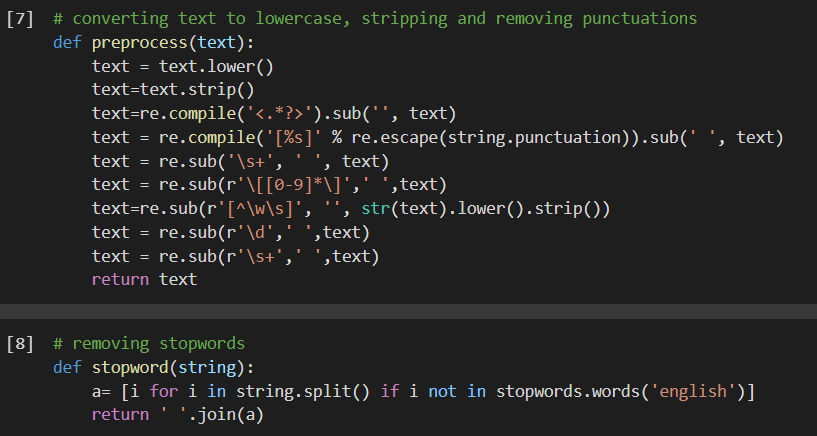
Now we simply feed this tf-idf representation (in the form of a matrix) and the class-list, i.e. Genre List to our model for it to train.

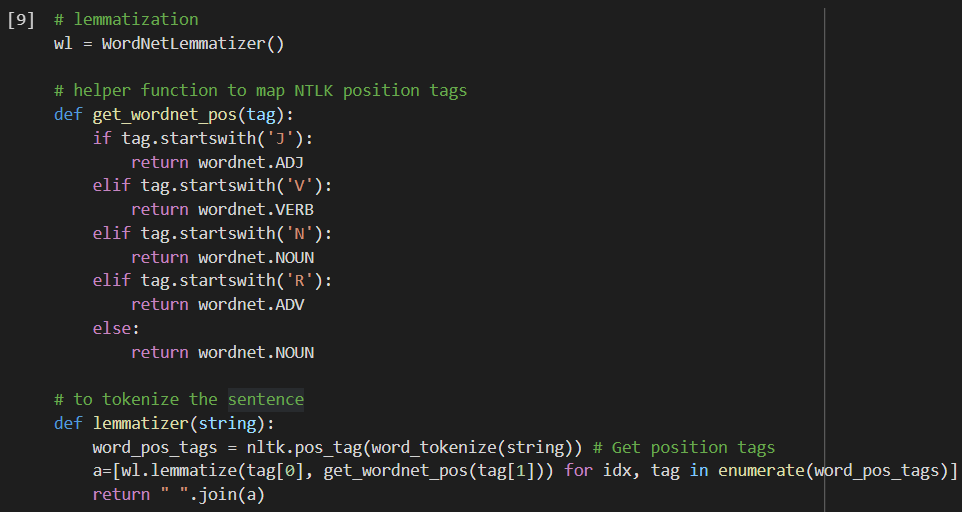
**FLOW OF PIPELINE**



**SCREENSHOTS**

Preprocessing Section:

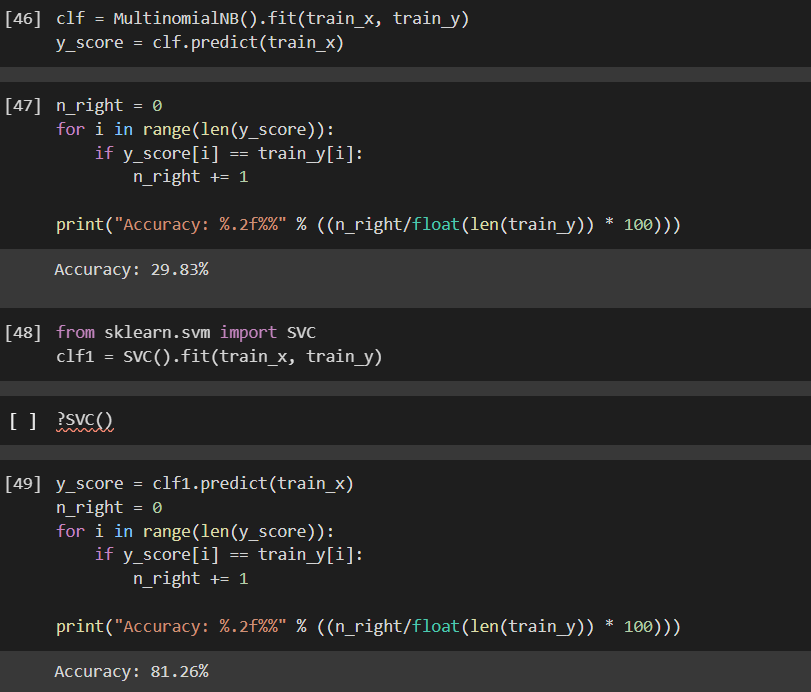




Dataset fed to the model:

****

Models tested, along with their accuracies:

****

**REFERENCES**

1. **Python Official Site:** [**https://www.python.org**](https://www.python.org)
2. **Elaboration on NLP: https://machinelearningmastery.com/natural-language-processing/**
3. **Wordnet Official Site: https://wordnet.princeton.edu/**
4. **Sklearn Documentation: https://scikit-learn.org/stable/index.html**
5. **IMDB for scraping training Data: https://www.imdb.com/**