|  |  |
| --- | --- |
| **Department of Computer Engineering**  **Class: B.E. (Computer) (Div- B) (Sem-VIII)** | |
| **Subject: Computational Lab-II (NLP)(CSL804)** | |
| **Sr. No.** | **Title of Experiment** |
| **5.** | **To perform POS tagging for any sample text.** |

|  |
| --- |
| **Course Name:** Computational Lab-II(NLP) |
| **Course Code:**  CSL804 |
| **Experiment No.:** 05 |
| **Lab outcome:** Acquire practical knowledge within the chosen area of technology for project development. |
| **Name of Student:** Rahul S Mahanty |
| **Student Roll No.:**214 |
| **Year/Semester/Div:** B.E./VIII/B |

**Experiment No. 05**

**Aim:** To perform POS tagging for any sample text.

**Theory:**

Natural language processing is a branch of machine learning that deals with how machines understand human languages. Text data is a widely available problem domain for NLP tasks. In order to work with text data, it is important to transform the raw text into a form that can be understood and used by machine learning algorithms, this is called text preprocessing. We have various techniques for text preprocessing like stemming, lemmatization, POS tagging, and dependency parsing.

**POS tags:**

Parts of speech tags are the properties of the words, which define their main context, functions, and usage in a sentence.

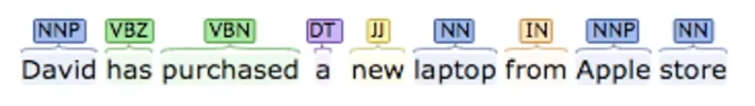


Some of the commonly used parts of speech tags are as follows:

**Nouns:** Which defines any object or entity

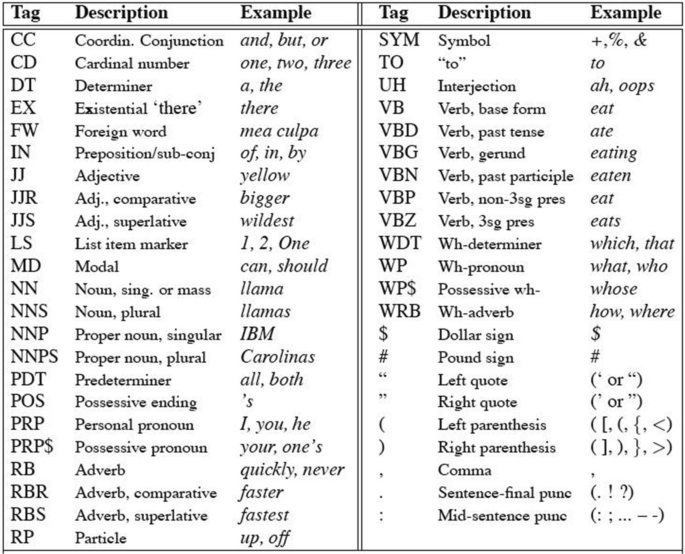
**Verbs:** That defines some action

**Adjectives and Adverbs:** This acts as a modifier, quantifier, or intensifier in any sentence.



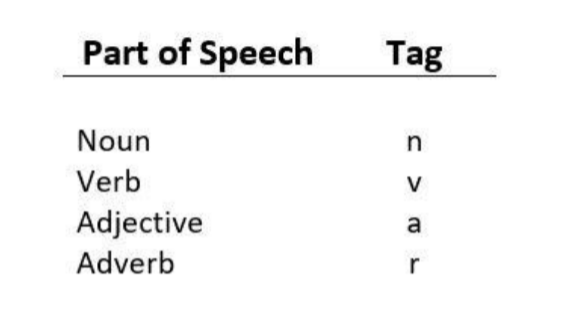
In this sentence, every word is associated with a part of the speech tag which defines their functions. Here, David has an NNP tag which means it is a proper noun. Further, Has and purchased belong to the verb indicating that they are the actions. The Laptop and Apple store are the nouns. New is the adjective whose role is to modify the context of the laptop.

We can apply machine learning models and rule-based models to obtain the parts of speech tags of a word. The most commonly used parts of speech tag notations are provided by the Penn Treebank corpus. In which, a total of 48 P.O.S tags are defined according to their usage.

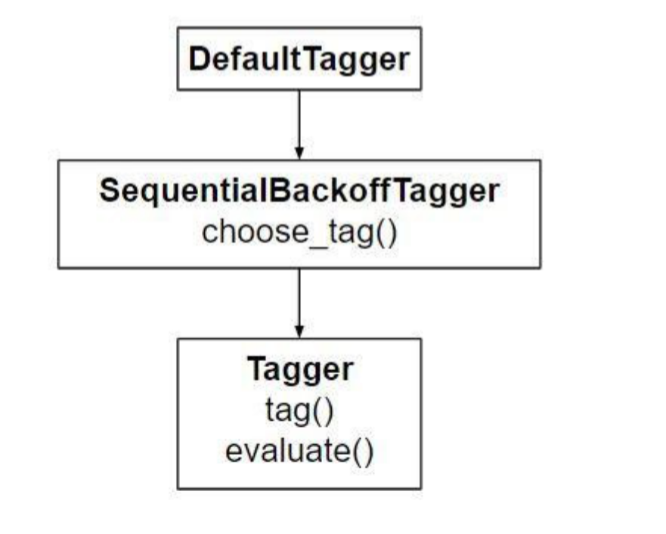


**POS Tagging:**

A process of converting a sentence to forms – list of words, list of tuples (where each tuple is having a form (word, tag)). The tag in case of is a part-of-speech tag, and signifies whether the word is a noun, adjective, verb, and so on.



**Default tagging** is a basic step for the part-of-speech tagging. It is performed using the Default\_Tagger class. The Default\_Tagger class takes ‘tag’ as a single argument. NN is the tag for a singular noun. Default \_Tagger is most useful when it gets to work with most common part-of-speech tag. that’s why a noun tag is recommended.



**POS tagging post Stop word removal:**

POS tagging is performed as sequence classification, so changing the sequence by removing stopwords will very likely change the POS tags for the remaining words. POS taggers are typically trained on complete text including stopwords. However after Removing stop words first will lose information for the POS tagger which is not something one would want for the language modelling.

**POS tagging post-split operation:**

Basically, the goal of a POS tagger is to assign linguistic (mostly grammatical) information to sub-sentential units. Such units are called tokens and, most of the time, correspond to words and symbols (e.g. punctuation).Thus split operation corresponds to like tokenization operation where each word in a sentence is spilt into an individual word like tokens. Thus, Now POS Tagging operation can be performed.

**Code #1 : How it works ?**

# Loading Libraries

from nltk.tag import DefaultTagger

# Defining Tag

tagging = DefaultTagger(“NN”)

# Tagging

tagging.tag([“Hello”, “Geeks”])

Output :

[(“Hello” ,“NN”), (“Geeks”, “NN”)]

Each tagger has a tag() method that takes a list of tokens (usually list of words produced by a word tokenizer), where each token is a single word. tag() returns a list of tagged tokens – a tuple of (word, tag).

**Program:**

import nltk

from nltk.tokenize import word\_tokenize

data ="All your dreams can come true if we have the courage to pursue them."

tokens= nltk.word\_tokenize(data)

tokens

wordsList = nltk.word\_tokenize(data)

tagged = nltk.pos\_tag(wordsList)

print(tagged)

import nltk

from nltk.corpus import stopwords

from nltk.tokenize import word\_tokenize, sent\_tokenize

stop\_words = set(stopwords.words('english' ))

txt = "Nothing is good or bad but thinking makes it so."

tokenized = sent\_tokenize(txt)

for i in tokenized:

wordsList = nltk.word\_tokenize(i)

wordsList = [w for w in wordsList if not w in stop\_words]

tagged = nltk.pos\_tag(wordsList)

print(tagged)

from nltk import pos\_tag

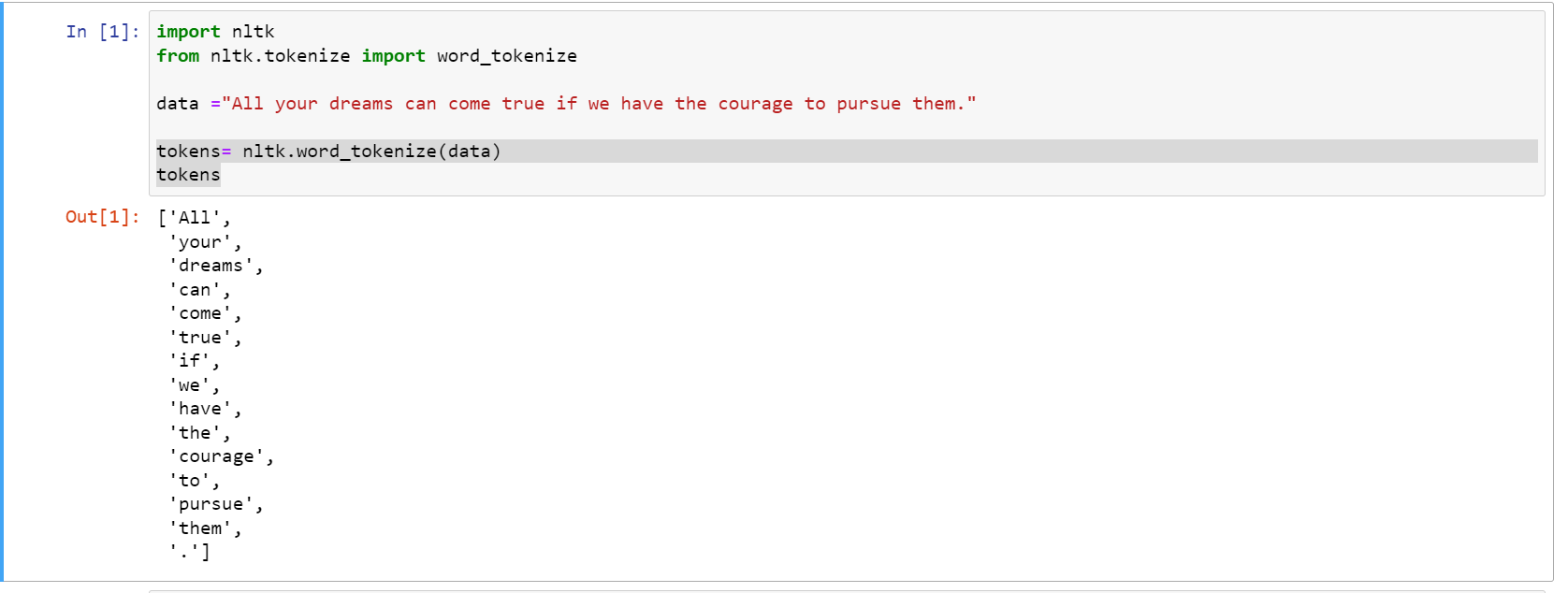
text =" Natural lanuage processing is an essential component in Language modelling".split()

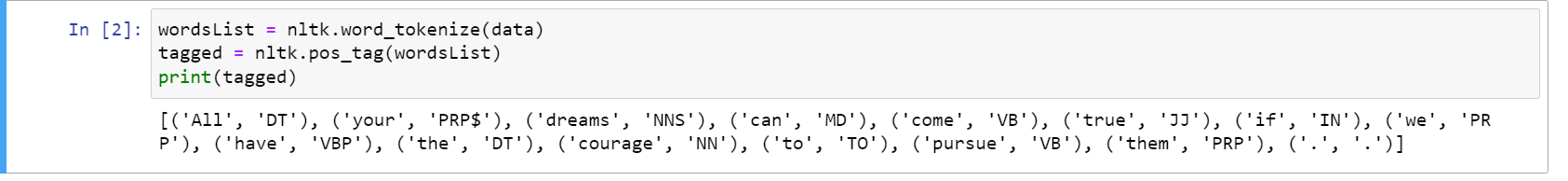
print("After split:",text)

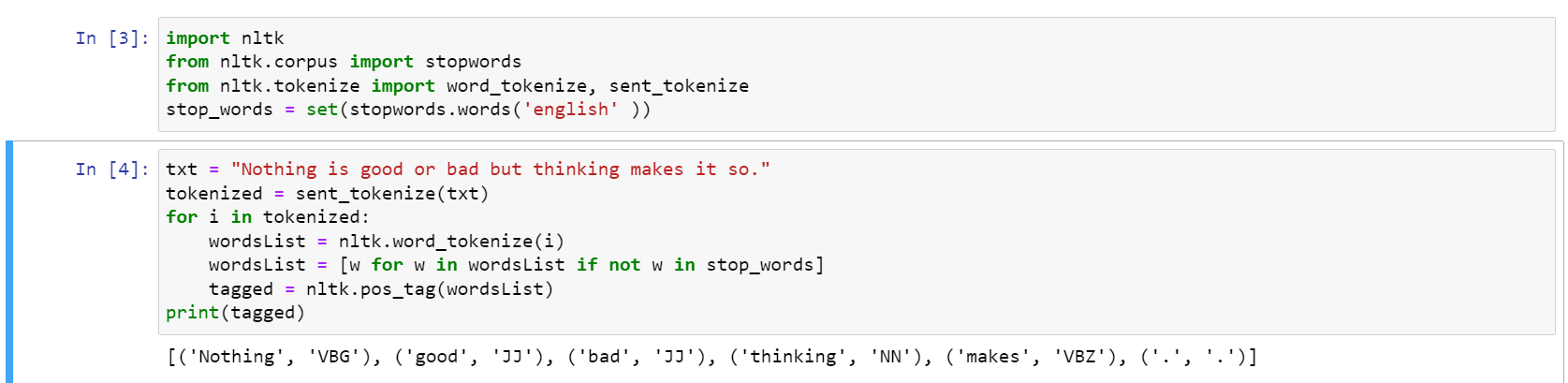
tokens\_tag =pos\_tag(text)

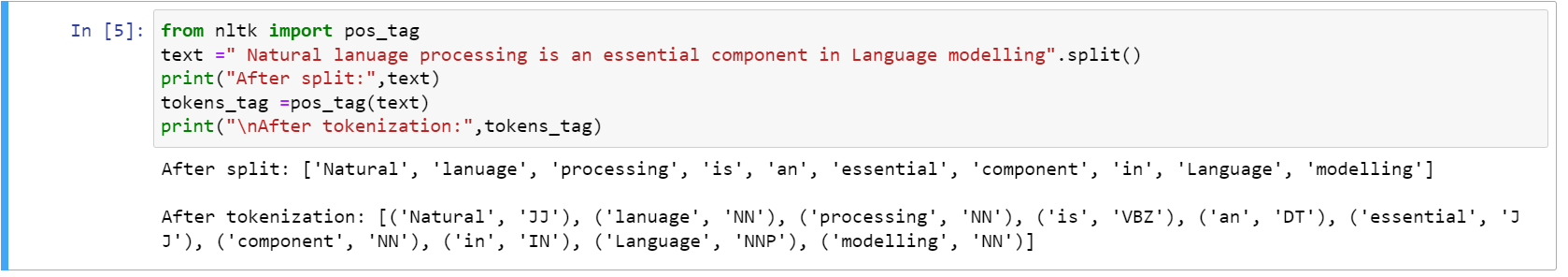
print("\nAfter tokenization:",tokens\_tag)

**Output:**

****

****

****

****

**Conclusion:** Henceforth, we have successfully performed POS tagging for any sample input text.