DIPTI SJARNAGAT CSEDS 55

Experiment No.1
Study various applications of NLP and Formulate the Problem Statement for Mini Project based on chosen real world NLP applications
Date of Performance:
Date of Submission:

Aim: Study various applications of NLP and Formulate the Problem Statement for Mini Project based on chosen real world NLP applications.

Objective: Understand the different applications of NLP and their techniques by reading and critiquing IEEE/ACM/Springer papers.

Theory:

1. **Machine Translation**

Machine translation is a process of converting the text from one language to the other automatically without or minimal human intervention.

2. **Text Summarization**

Condensing a lengthy text into a manageable length while maintaining the essential informational components and the meaning of the content is known as summarization. Since manually summarising material requires a lot of time and is generally difficult, automating the process is becoming more and more popular, which is a major driving force behind academic research.

Text summarization has significant uses in a variety of NLP-related activities, including text classification, question answering, summarising legal texts, summarising news, and creating headlines. Additionally, these systems can incorporate the creation of summaries as a middle step, which aids in shortening the text.

The quantity of text data from many sources has multiplied in the big data era. This substantial body of writing is a priceless repository of data and expertise that must be skillfully condensed in order to be of any use. A thorough investigation of NLP for automatic text summarization has been necessitated by the increase in the availability of documents. Automatic text summarising is the process of creating a succinct, fluid summary without the assistance of a human while maintaining the original text's meaning.

3. Sentiment Analysis

Sentiment analysis, often known as opinion mining, is a technique used in natural language processing (NLP) to determine the emotional undertone of a document. This is a common method used by organisations to identify and group ideas regarding a certain good, service, or concept. Text is mined for sentiment and subjective information using data mining, machine learning, and artificial intelligence (AI).

Opinion mining can extract the subject, opinion holder, and polarity (or the degree of positivity and negative) from text in addition to identifying sentiment. Additionally, other scopes, including document, paragraph, sentence, and sub-sentence levels, can be used for sentiment analysis.

Businesses must comprehend people's emotions since consumers can now communicate their views and feelings more freely than ever before. Brands are able to listen carefully to their customers and customise their products and services to match their demands by automatically evaluating customer input, from survey replies to social media chats.

4. Information Retrieval

A software programme that deals with the organisation, storage, retrieval, and evaluation of information from document repositories, particularly textual information, is known as information retrieval (IR). The system helps users locate the data they need, but it does not clearly return the questions' answers. It provides information about the presence and placement of papers that may contain the necessary data. Relevant documents are those that meet the needs of the user. Only relevant documents will be pulled up by the ideal IR system.

5. Question Answering System (QAS)

Building systems that automatically respond to questions presented by humans in natural language is the focus of the computer science topic of question answering (QA), which falls under the umbrella of information retrieval and natural language processing (NLP).

OUTPUT:-

TITLE NAME :- "AUTO RECOMMENDATION TEXT ENTRY"

Conclusion:

The Auto Recommendation Text Entry system is a promising solution to address the challenges of efficient and accurate text input in NLP and other text-based applications. With its real-time word suggestions and user-friendly GUI, it aims to improve productivity and reduce typing errors. This report has outlined the problem statement, objectives, scope, and introduced the proposed system, setting the stage for its potential applications in a variety of task.

DIPTI SHARNAGAT CSE DS 55

Experiment No.2
Apply Tokenization on given English and Indian Language Text
Date of Performance:
Date of Submission:

Aim: Apply Tokenization on given English and Indian Language Text

Objective: Able to perform sentence and word tokenization for the given input text for English and Indian Langauge.

Theory:

Tokenization is one of the first step in any NLP pipeline. Tokenization is nothing but splitting the raw text into small chunks of words or sentences, called tokens. If the text is split into words, then its called as 'Word Tokenization' and if it's split into sentences then its called as 'Sentence Tokenization'. Generally 'space' is used to perform the word tokenization and characters like 'periods, exclamation point and newline char are used for Sentence Tokenization. We have to choose the appropriate method as per the task in hand. While performing the tokenization few characters like spaces, punctuations are ignored and will not be the part of final list of tokens.

Why Tokenization is Required?

Every sentence gets its meaning by the words present in it. So by analyzing the words present in the text we can easily interpret the meaning of the text. Once we have a list of words we can also use statistical tools and methods to get more insights into the text. For example, we can use word count and word frequency to find out important of word in that sentence or document.

Input Text

Tokenization is one of the first step in any NLP pipeline. Tokenization is nothing but splitting the raw text into small chunks of words or sentences, called tokens.

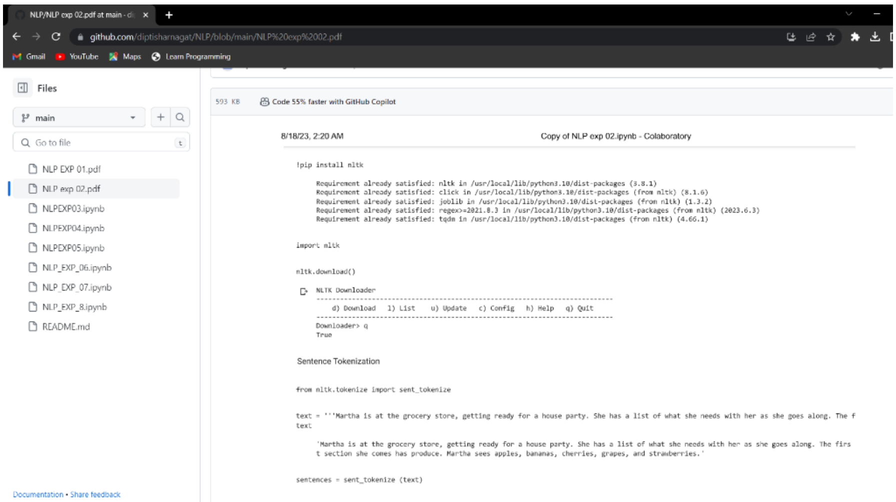
Word Tokenization

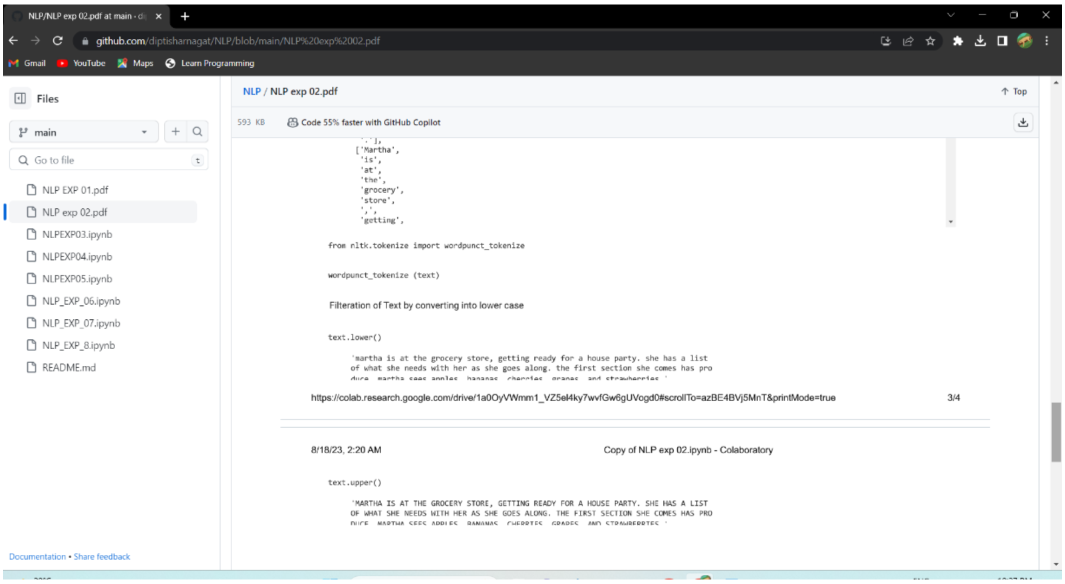
Tokenization	is	one	of
the	first	step	in
any	NLP	pipeline	Tokenization
is	nothing	but	splitting
the	raw	text	into
small	chunks	of	words
or	sentences	called	tokens

Sentence Tokenization

Tokenization is one of the first step in any NLP pipeline

Tokenization is nothing but splitting the raw text into small chunks of words or sentences, called tokens





Conclusion:

okenization is the process of breaking down the given text in natural language processing into the smallest unit in a sentence called a token. Punctuation marks, words, and numbers can be considered tokens.

DIPTI SHARNAGAT CSEDS 55

Experiment No.3
Apply Stop Word Removal on given English and Indian Language Text
Date of Performance:
Date of Submission:

Aim: Apply Stop Word Removal on given English and Indian Language Text.

Objective: To write program for Stop word removal from a sentence given in English and any Indian Language.

Theory:

The process of converting data to something a computer can understand is referred to as pre-processing. One of the major forms of pre-processing is to filter out useless data. In natural language processing, useless words (data), are referred to as stop words.

Stopwords are the most common words in any natural language. For the purpose of analyzing text data and building NLP models, these stopwords might not add much value to the meaning of the document.

Stop Words: A stop word is a commonly used word (such as "the", "a", "an", "in") that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query. We need to perform tokenization before removing any stopwords.

Why do we need to Remove Stopwords?

Removing stopwords is not a hard and fast rule in NLP. It depends upon the task that we are working on. For tasks like text classification, where the text is to be classified into different categories, stopwords are removed or excluded from the given text so that more focus can be given to those words which define the meaning of the text.

Here are a few key benefits of removing stopwords:

- On removing stopwords, dataset size decreases and the time to train the model also decreases
- Removing stopwords can potentially help improve the performance as there are fewer and only meaningful tokens left. Thus, it could increase classification accuracy

- Even search engines like Google remove stopwords for fast and relevant retrieval of data from the database

We can remove stopwords while performing the following tasks:

- Text Classification
    - Spam Filtering
    - Language Classification
    - Genre Classification
- Caption Generation
- Auto-Tag Generation

Avoid Stopword Removal

- Machine Translation
- Language Modeling
- Text Summarization
- Question-Answering problems

Different Methods to Remove Stopwords

1. **Stopword Removal using NLTK**
   NLTK, or the Natural Language Toolkit, is a treasure trove of a library for text preprocessing. It's one of my favorite Python libraries. NLTK has a list of stopwords stored in 16 different languages.

   You can use the below code to see the list of stopwords in NLTK:

```
import nltk
from nltk.corpus import stopwords
set(stopwords.words('english'))
```

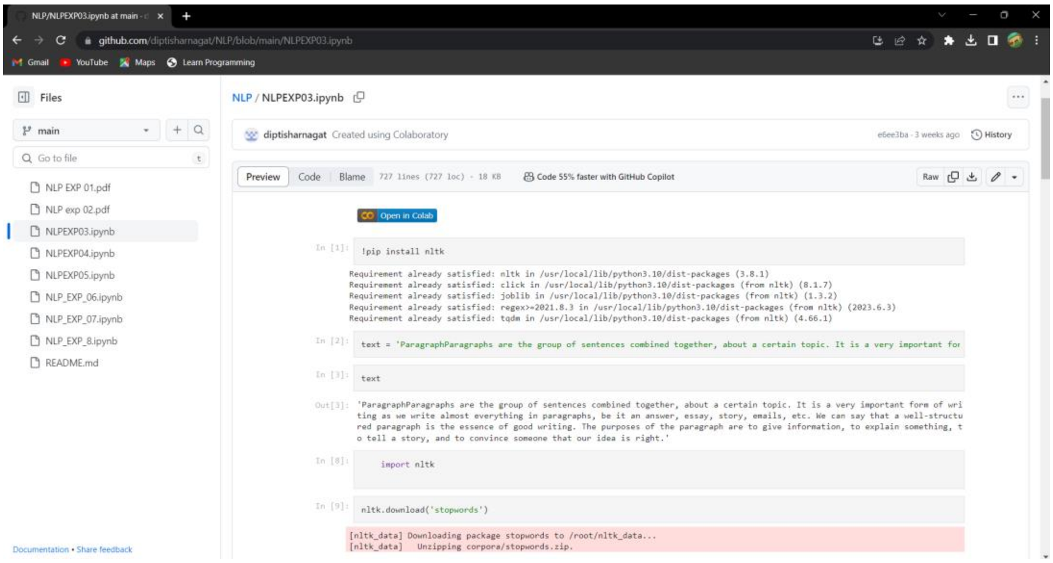
2. **Stopword Removal using spaCy:**

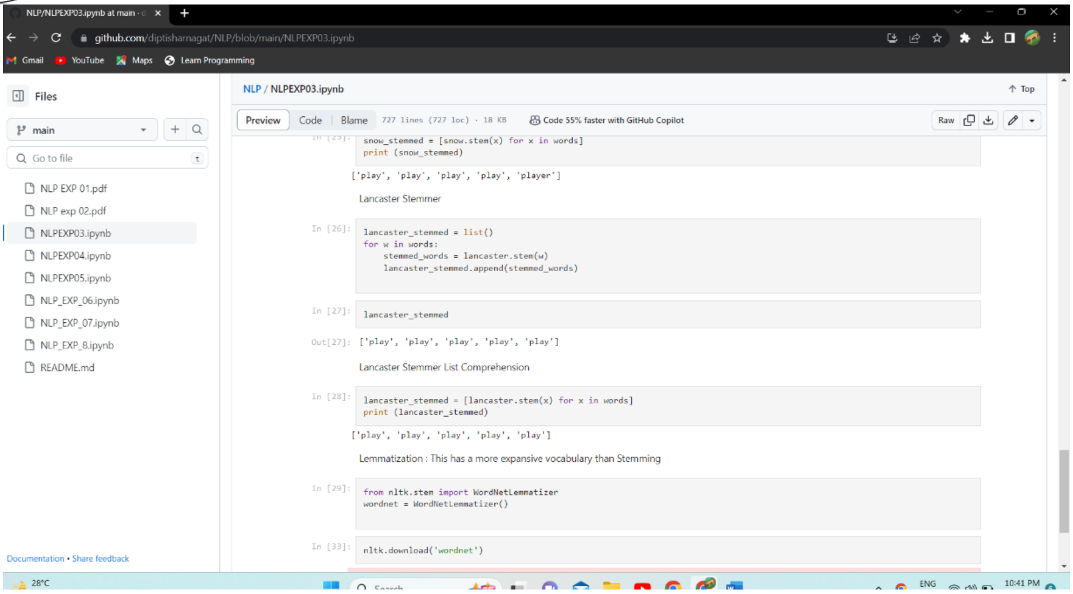
   spaCy is one of the most versatile and widely used libraries in NLP. We can quickly and efficiently remove stopwords from the given text using SpaCy.

   It has a list of its own stopwords that can be imported as **STOP_WORDS** from the **spacy.lang.en.stop_words** class.

3. **Stopword Removal using Gensim**

   Gensim is a pretty handy library to work with on NLP tasks. While pre-processing, gensim provides methods to remove stopwords as well. We can easily import the remove_stopwords method from the class gensim.parsing.preprocessing.





Conclusion:

- We provide a brief description about experimental setup to see the effect of stopwords in the Indian languages Marathi, Bengali, Gujarati and Sanskrit. Stopwords are the most frequently used words like articles, pronouns, conjunctions, prepositions, prefixes, adverbs and adjectives.

DIPTI SHARNAGAT CSEDS 55

Experiment No.4
Apply Stemming on the given Text input
Date of Performance:
Date of Submission:

Aim: Apply Stemming on the given Text input.

Objective: Understand the working of stemming algorithms and apply stemming on the given input text.

Theory:

Stemming is a process of linguistic normalization, which reduces words to their word root word or chops off the derivational affixes. For example, connection, connected, connecting word reduce to a common word "conect".

Stemming is the process of producing morphological variants of a root/base word. Stemming programs are commonly referred to as stemming algorithms or stemmers. A stemming algorithm reduces the words "chocolates", "chocolatey", "choco" to the root word, "chocolate" and "retrieval", "retrieved", "retrieves" and reduces to the stem "retrieve". Stemming is an important part of the pipelining process in Natural language processing. The input to the stemmer is tokenized words.

Applications of stemming :

1. Stemming is used in information retrieval systems like search engines.
2. It is used to determine domain vocabularies in domain analysis.

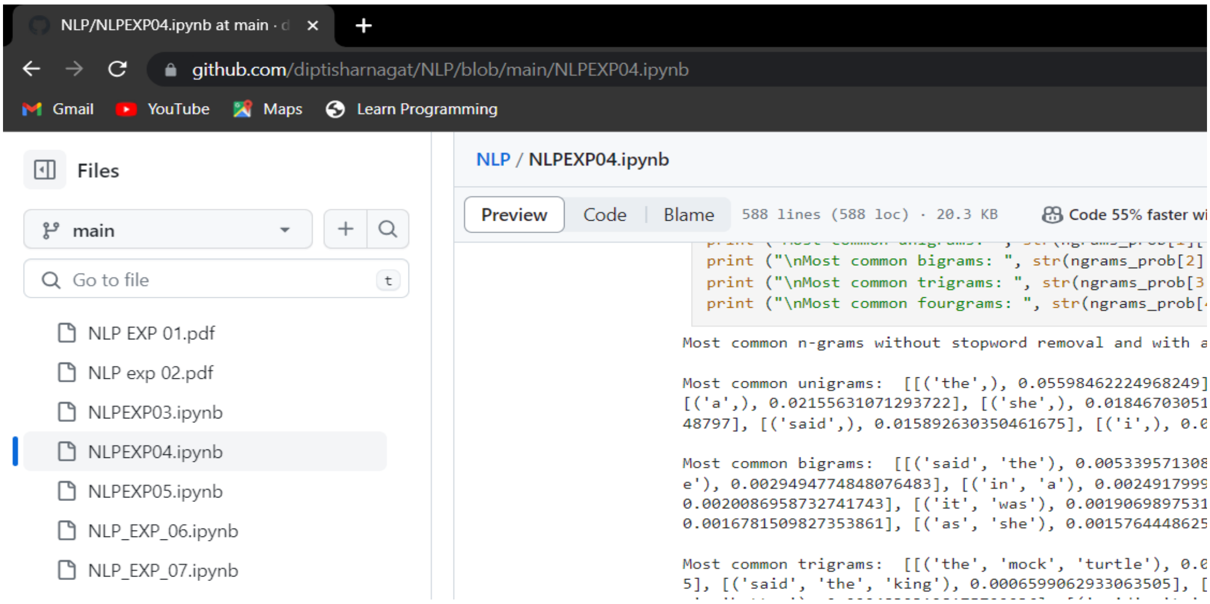
Porter's Stemmer Algorithm:

It is one of the most popular stemming methods proposed in 1980. It is based on the idea that the suffixes in the English language are made up of a combination of smaller and simpler suffixes. This stemmer is known for its speed and simplicity. The main applications of Porter Stemmer include data mining and Information retrieval. However, its applications are only limited to English words. Also, the group of stems is mapped on to the same stem and the output stem is not necessarily a meaningful word. The algorithms are fairly lengthy in nature and are known to be the oldest stemmer.

Example: EED -> EE means "if the word has at least one vowel and consonant plus EED ending, change the ending to EE" as 'agreed' becomes 'agree'.

Advantage: It produces the best output as compared to other stemmers and it has less error rate.

Limitation: Morphological variants produced are not always real words.



- **Conclusion:** Stemming is a natural language processing technique that is used to reduce words to their base form, also known as the root form. The process of stemming is used to normalize text and make it easier to process. It is an important step in text pre-processing, and it is commonly used in information retrieval and text mining applications.

DIPTI SHARNAGAT CSEDS 55

Experiment No.5
Implement Bi-Gram model for the given Text input
Date of Performance:
Date of Submission:

Aim: Implement Bi-Gram model for the given Text input

Objective: To study and implement N-gram Language Model.

Theory:

A language model supports predicting the completion of a sentence.
Eg:

- Please turn off your cell _____
- Your program does not ______

Predictive text input systems can guess what you are typing and give choices on how to complete it.

N-gram Models:
Estimate probability of each word given prior context.
P(phone | Please turn off your cell)

- Number of parameters required grows exponentially with the number of words of prior context.
- An N-gram model uses only N1 words of prior context.
    - Unigram: P(phone)
    - Bigram: P(phone | cell)
    - Trigram: P(phone | your cell)

- The Markov assumption is the presumption that the future behavior of a dynamical system only depends on its recent history. In particular, in a kth-order Markov model, the next state only depends on the k most recent states, therefore an N-gram model is a (N1)-order Markov model.

N-grams: a contiguous sequence of n tokens from a given piece of text

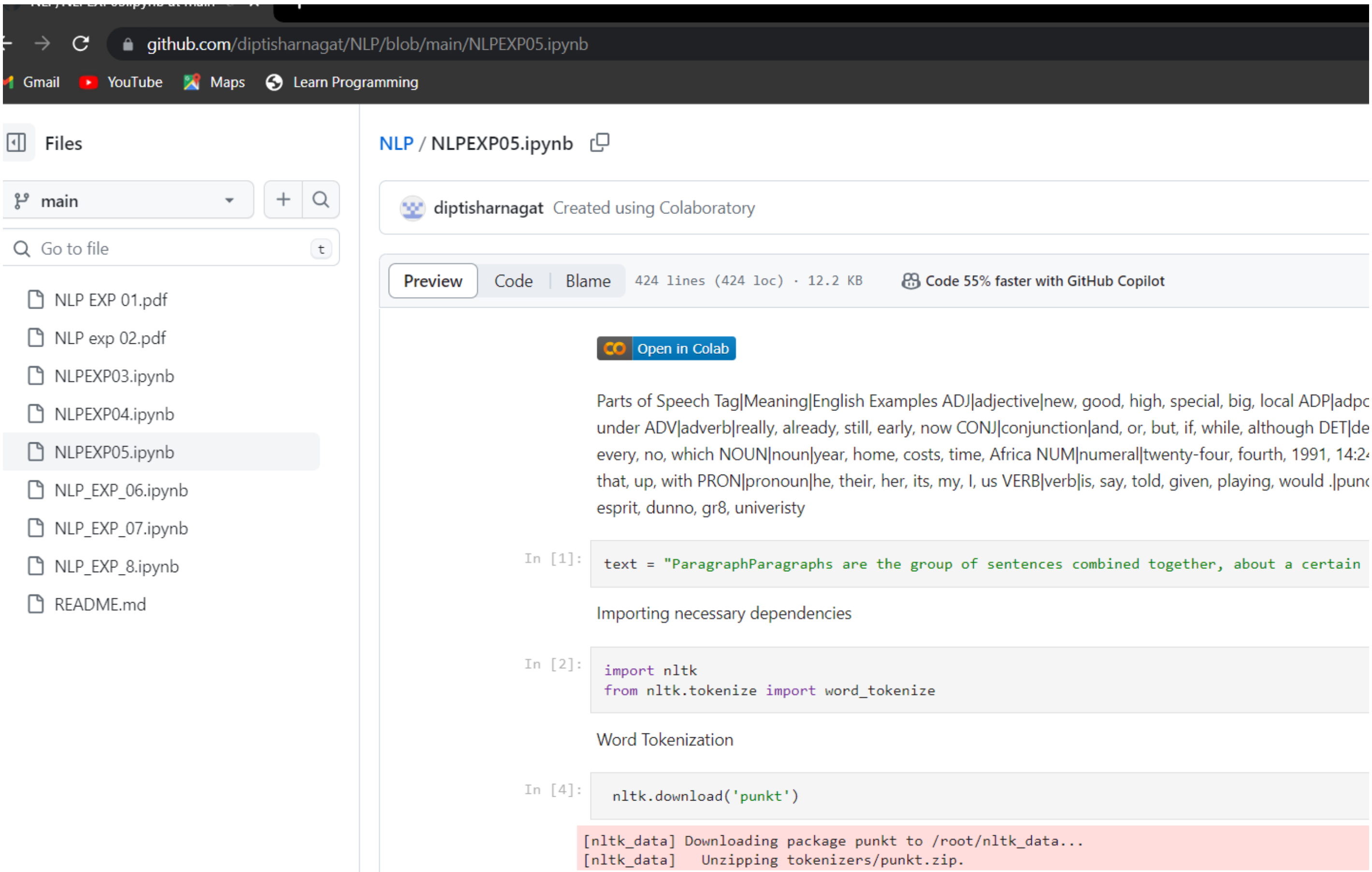


Fig. Example of Trigrams in a sentence

Conclusion:

he model implemented here is a "Statistical Language Model". I have used "BIGRAMS" so this is known as Bigram Language Model. In Bigram language model we find bigrams which means two words coming together in the corpus (the entire collection of words/sentences). In the sentence "DEV is awesome and user friendly" the bigrams are :

DIPTI SHARNAGAT CSEDS 55

Experiment No.6
Perform POS tagging on the given English and Indian Language Text
Date of Performance:
Date of Submission:

Aim: Perform POS tagging on the given English and Indian Language Text

Objective: To study POS Tagging and tag the part of speech for given input in english and an Indian Language.

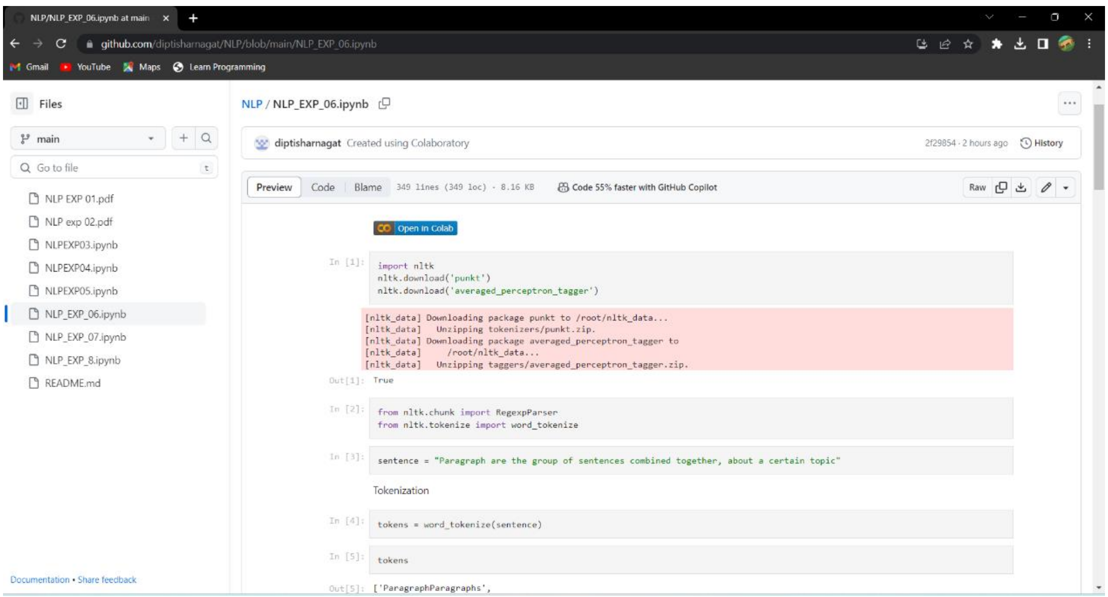
Theory:

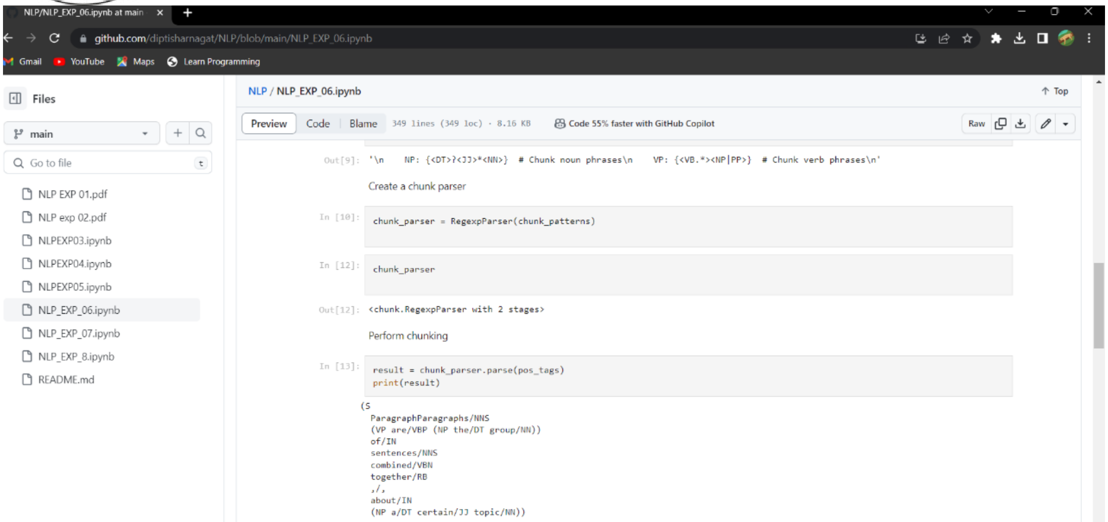
The primary target of Part-of-Speech (POS) tagging is to identify the grammatical group of a given word. Whether it is a NOUN, PRONOUN, ADJECTIVE, VERB, ADVERBS, etc. based on the context. POS Tagging looks for relationships within the sentence and assigns a corresponding tag to the word.

POS Tagging (Parts of Speech Tagging) is a process to mark up the words in text format for a particular part of a speech based on its definition and context. It is responsible for text reading in a language and assigning some specific token (Parts of Speech) to each word. It is also called grammatical tagging.

Steps Involved in the POS tagging example:

- Tokenize text (word_tokenize)
- apply pos_tag to above step that is nltk.pos_tag(tokenize_text)





Conclusion:

POS Tagging (Parts of Speech Tagging) is a process to mark up the words in text format for a particular part of a speech based on its definition and context.

DIPTI SHARNAGAT CSEDS 55

Experiment No.7
Implement Named Entity Recognizer for the given Text input
Date of Performance:
Date of Submission:

Aim: Implement Named Entity Recognizer for the given Text input

Objective: Understand the importance of NER in NLP and Implement NER.

Theory:

The named entity recognition (NER) is one of the most data preprocessing task. It involves the identification of key information in the text and classification into a set of predefined categories. An entity is basically the thing that is consistently talked about or refer to in the text.

NER is the form of NLP.

At its core, NLP is just a two-step process, below are the two steps that are involved:

- Detecting the entities from the text
- Classifying them into different categories

Some of the categories that are the most important architecture in NER such that:

- Person
- Organization
- Place/ location

Other common tasks include classifying of the following:

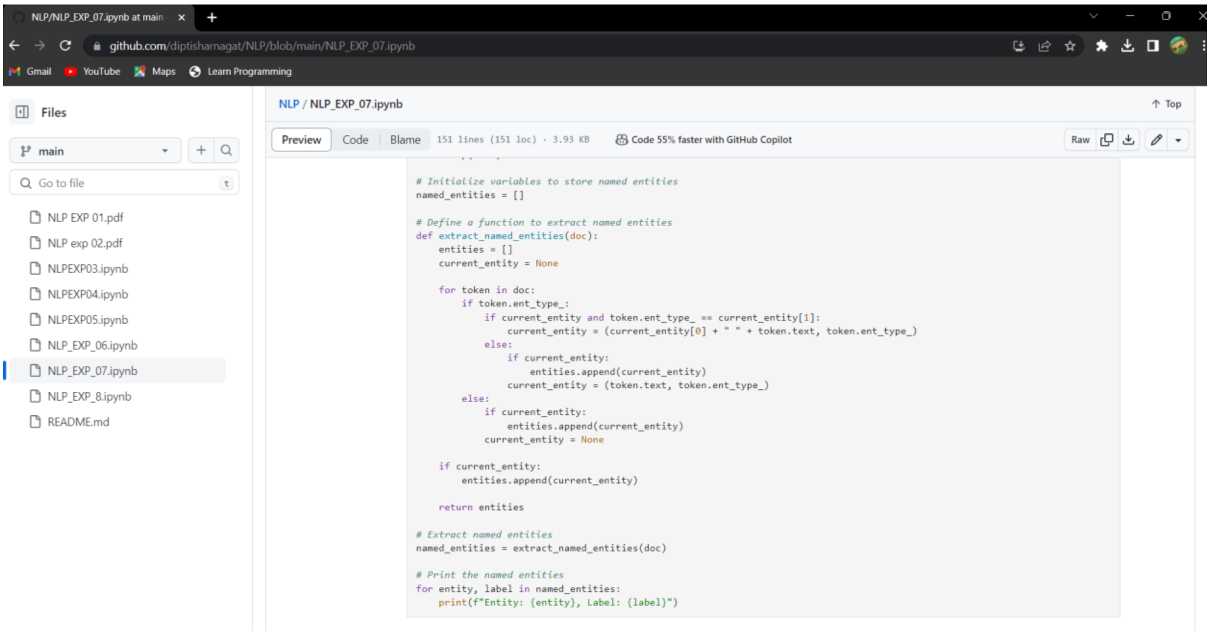
- date/time.
- expression
- Numeral measurement (money, percent, weight, etc)
- E-mail address

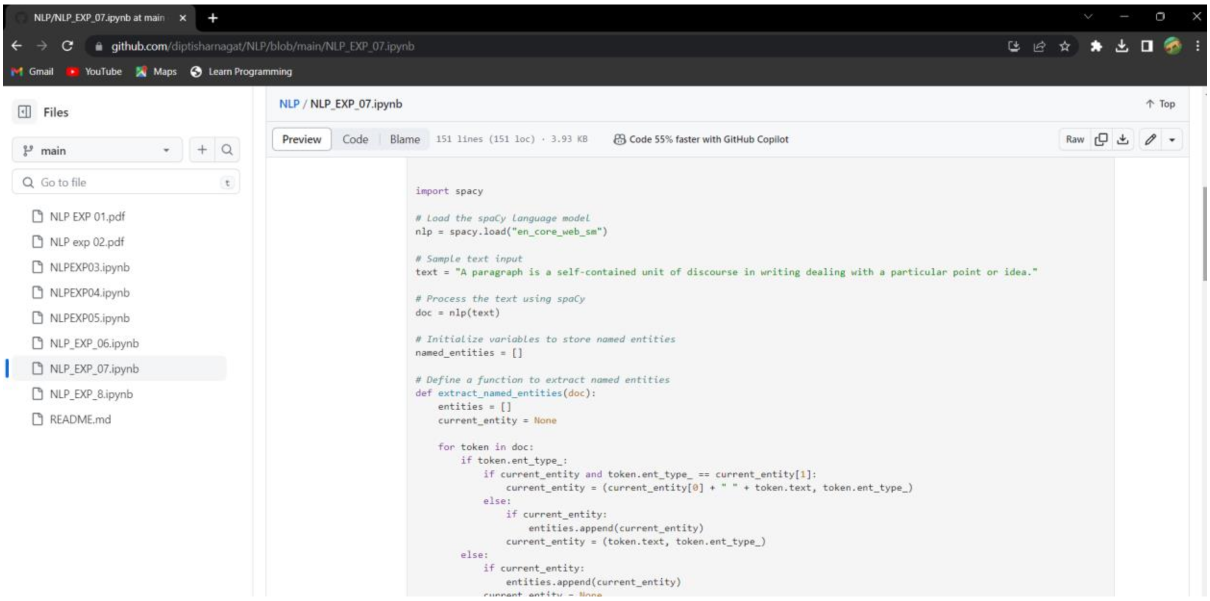
Ambiguity in NE

For a person, the category definition is intuitively quite clear, but for computers, there is some ambiguity in classification. Let's look at some ambiguous example:

England (Organisation) won the 2019 world cup vs The 2019 world cup happened in England(Location).

Washington(Location) is the capital of the US vs The first president of the US was Washington(Person).





Conclusion:

A named entity is basically a real-life object which has proper identification and can be denoted with a proper name. Named Entities can be a place, person, organization, time, object, or geographic entity. For example, named entities would be Roger Federer, Honda city, Samsung Galaxy S10. Named entities are usually instances of entity instance

Experiment No. 8
Implement word sense disambiguation using LSTM/GRU

DIPTI

Date of Performance:	
Date of Submission:	

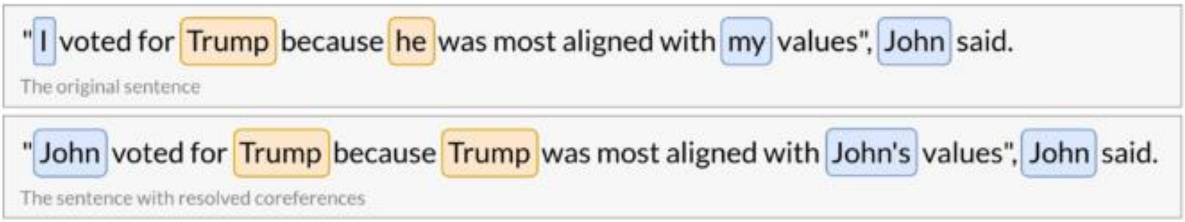
SHARNAGAT CSEDS 55

Aim: Apply Reference Resolution Technique on the given Text input.

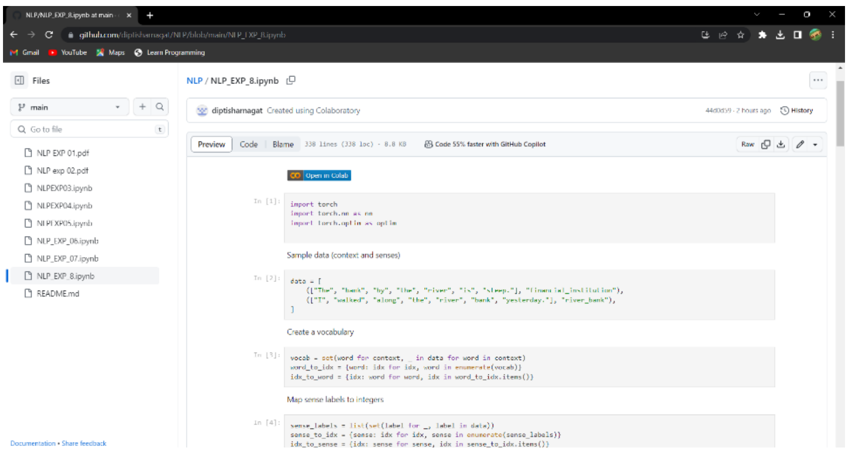
Objective: Understand the importance of resolving references and implementing reference resolution for the given text input.

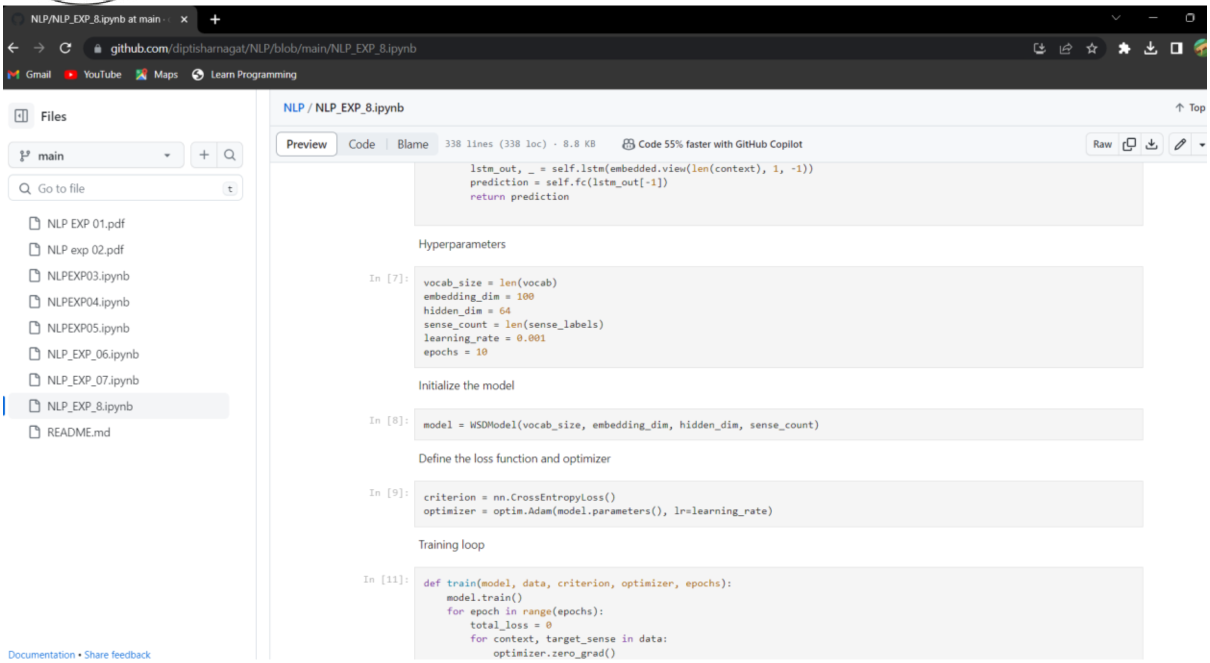
Theory:

Coreference resolution (CR) is the task of finding all linguistic expressions (called mentions) in a given text that refer to the same real-world entity. After finding and grouping these mentions we can resolve them by replacing, as stated above, pronouns with noun phrases.



Coreference resolution is an exceptionally versatile tool and can be applied to a variety of NLP tasks such as text understanding, information extraction, machine translation, sentiment analysis, or document summarization. It is a great way to obtain unambiguous sentences which can be much more easily understood by computers.





Conclusion:

Word Sense Disambiguation is considered one of the challenging problems in natural language processing (NLP). LSTM-based Word Sense Disambiguation techniques have been shown effective through experiments. Models have been proposed before that employed LSTM to achieve state-of-the-art results.