1. **What all problem you have solved in NLP. Give some examples.**

Ans- In NLP I have worked on the sentiment analysis of reviews of institutions. The requirement was to classify the reviews for institutions in good or bad or average.

I used NLTK to process the reviews and removed all the words or character like “I”, “you” and punctuations etc and created a feature of reviews with words which we could use for preparing the proper feature columns for our model.

I also used word cloud to show the words and their frequency for all classes. Next with the help of word to vector I prepared the feature column and then I applied these to different Machine Learning algorithms and calculated all the accuracy scores, confusion matrix, Precision and recall to select the best model which resulted that the random forest model was best suitable for it. So I finalised it for final predicting model.

1. **What is NLP?**

NLP (Natural Language Processing) is a Artificial Intelligence method or a branch of it which is used to make system understand the natural languages like English in a way we humans understand it, the problem with the computer system is that does not understand text it can convert a string to ASCII and then to binary but it cannot interpret what words means and what is the context behind that word. To solve this issue we use NLP which has many algorithms, libraries and methods (NLTK, Word 2 vector model, TFIDF) where system tries to understand the context and meaning through mathematical relationships between words. The approach of NLP is that first it pre-processes the data to remove unwanted words or character or punctuations (“the”,”.”) and then convert all the words with affixes like “ing”,”es” or words with similar meanings to its root words, this gives a set of words which are actually useful in finding the context of the text, after this all these words are converted into its numerical representation using various techniques like Bag of Words, TFIDF and Word2Vec , because ultimately the system tries to build a mathematical relationship among them and so we need to give numeric data for it. Then this numeric output is given to various Machine Learning and Deep Learning algorithms to predict any value out of these input or we can also classify these into different classes like good or bad sentiment etc.

One of the common uses of it is in analysing sentiments from user reviews about a product or about movie. It is also widely use for making chatbots where system tries to understand the questions of user and provide them appropriate answers just like humans.

1. **Libraries you have used for NLP.**

I have used several libraries in NLP, like-

NLTK (Natural Language Tool Kit)-This is one of the most important library in NLP, the main use of the NLTK is in extracting useful words from a text. A sentence in natural language contains various parts like punctuations (‘,’,’.’), article words (‘the’,’an’,a’). These words may be useful in framing a proper sentence but these does not have any impact in understanding the context and sentiment out of the text, In NLTK we have several features (like stopwords) which can identify these words and helps in removing these. In language we have several forms of words based on tense like talk, taking gone, go these words have same meaning so NLTK provide to methods stemming and lemmatization which can be used to find root words of all.

**Textblob-**

This is another important library, it is based on the NLTK itself but one important advantage of it is that it is more simpler than NLTK , and it can be termed as interface for NLTK.

**spaCy-**

This is very light and fast library, it is written in Cython (It is C Extension of python) ,it does not provide too many options like NLTK ,it works on principle of one effective algorithm for one task. It has algorithms for POS tagging, tokenization.

1. **What do you understand by tokenization?**

Tokenization is the process of breaking up of stream of text into the words, phrases, symbols and other meaningful elements and these elements are known as tokens. It is one of the most important step for NLP, it gives the list of words which can be further processed to remove less important words (stop words) and punctuation and find the useful words out of these tokens.

1. **List down types of tokenization?**

There are mainly two types of tokenization –

* **Word Tokenizers (Treebank Tokenizer)–** Tokenization is the process of dividing a text into smaller unit called tokens which is nothing but series of character separated by space. Word tokenizer divides a text into tokens or a list of words.
* **Sentence Tokenization-** sentence tokenizer divides the text into a list of sentences , it is also known as sentence boundary detection it tries to detect end of sentence using punctuation mark and tart of next sentence , it sometimes ambiguous as punctuation can be used for different purpose too like 2.0 etc. The sentence tokenizer uses punktsentence tokenizer which an unsupervised algorithm which has been trained for many European languages,

1. **what do you understand by stemming?**

Stemming is the process of chopping of the affixes of words to find root words. In English grammar we have words written in different forms based on singular, plural and tenses. For example, flowers, going, running

So after stemming words will be like-

Flowers- flower

Going - go

goes - go

chocolates- chocolate

The issue with the stemmer is that it ignores that the word formed after stemming is having dictionary meaning or not.

There are two types of problems in it-

1. **Over stemming**- Over stemming occurs when stemming algorithm chop off too many characters which results in words which has no sense, this also occurs when two words are stemmed to same root words but actually, they belong to different stem or root words.

For example if we have four words as universal, universities, Universe and University, the most of the porter will stem in single word univers, this not the ideal stemming we should do the ideal stems will be Universe for Universal and Universe and university for Universities and University

1. **Under stemming**- This occurs when two words are stemmed to different stem or root words but actually, they belong to same stem.

There are various types of algorithms used for stemming as below-

1. **Porter stemmer-** It is the most commonstemmerwhich is widely used. It takes into consideration that suffixes in English language is smaller one. This is also consider as a moderate stemmer because it does not chop off all the suffixes deeply.
2. **Lancaster Stemmer-** It also chops of the suffixes and it is more aggressive than porter stemmer.
3. **Snowball Stemmer-** This is based on Snowball language, The advantage of it is that it’s a very moderate an gives more readable outputs. As it has support for many languages, we need supply a parameter to say for which language the stemming is going to be done.

7. **Bag of word, TF IDF, word to vector, doc to vector**

These are the techniques for feature extraction in NLP.As we know that the machine learning algorithms does not take text as input , in machine learning the model try to build a mathematical relationship between the features and label . So we need to convert the text data to numerical data. We can do it in several ways as below-

**Bag of Word-** Bag of word is on of the most simple feature extraction technique. In this first a vocabulary of all unique word or known words in all the documents are made .Now then a word vector is made for each document by taking 1 if a word is present an 0 if word is not present. Example

For words it is nice sunny day and its not raining.

So if a document has line- it is nice sunny day

Then for this document the vector will be-

It=1

Is=1

nice=1

sunny=1

day=1

and =0

its=0

not =0

raining=0

Word vector= 1 1 1 1 1 0 0 0 0

Bag of Word is only concern fact that if a word is present in a document or not, it does not take into account that how many times a word is present or in which order the word is present.

**TFIDF-** The issue with the bag of word is that it only consider if a word is present in a document or not. So eventually it gives similar importance to all the words, so the words like the, us, is was etc. Gets the same importance, these words can be important for making a sentence as per grammar rules but these are not useful for getting context out of it. To overcome this we can use TFIDF . TFIDF calculates the term frequency

TF(Term Frequency) = Number of times a term or a word is present in a document/ Number of terms in the document.

IDF(Inverse Document frequency)- It is calculated as-

Log(Total number of document/number document with key)

The above calculation normalizes the impact of words like the, is .Because these words will be present in almost all the documents and so

Total number of document = Number of document with key suppose 5

So IDF =log(5/5)=log(1)=0

And o TF\*IDF=TF\*0=0

So it normalizes the impact of common words, and it also gives importance to words which are occurring less number of times but has greater impact on context, in this case even though Term Frequency is low but multiplied value of TF and IDF will make the result significant till some extent.

So TFIDF reduces impact of common but not useful words and give significant importance to less common but useful words.

**Word2Vec-** This is a neural network algorithm model which we need to train with our vocabulary. It tries to find the context of the word based on corpus given to it; it places each word in a vector space and tries to find the word near to it. If want to add two words and find the next word (Apple + Orange= Fruit) or if we want to subtract one word from another then in these case the vectors are added or subtracted which will give another point in vector space and then it tries to find words near to this new vector.

To represent a word in vector form it uses concurrence matrix which represent the count of one word appearing next to other.

It uses two main algorithms, first is continuous bag words where context words are used to predict the target word, second algorithm is skip gram where context words are predicted using target word.

**Doc2Vec-** Doc2vec is based on the word to vector , Word2vec represent the concept of a word but the doc2vec represent the concept of document, in Doc2vec a extra vector is added which is unique id for each document so model is trained with this id also, and at the end this id holds the numeric representation of the document.

**8. Which algorithms you have used for NLP?**

In NLP I have used several algorithms at several stages, for feature extraction I have used Bag of Words, TFIDF and Word2vec. And to prepare model using regression I used logistic regression and classification I have used mostly Naive Bayes and Random Forest.