INTELLIPAAT

https://intellipaat.com/blog/interview-question/nlp-interview-questions/

Top 30 NLP Interview Questions and Answers

Basic NLP Interview Questions:

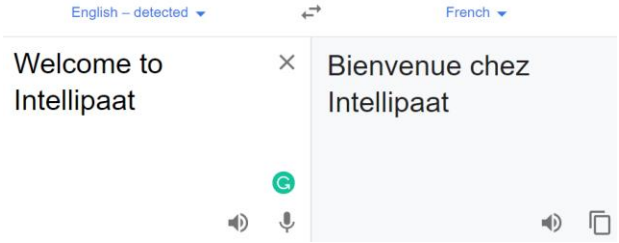
1. What do you understand by Natural Language Processing?

Natural Language Processing is a field of computer science that deals with communication between computer systems and humans. It is a technique used in Artificial Intelligence and Machine Learning. It is used to create automated software that helps understand human-spoken languages to extract useful information from the data. Techniques in NLP allow computer systems to process and interpret data in the form of natural languages

2. List any two real-life applications of Natural Language Processing.

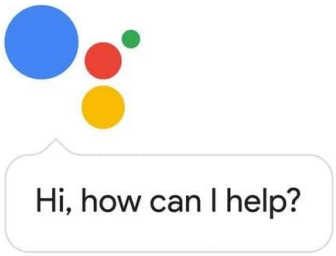
Two real-life applications of Natural Language Processing are as follows:

1. **Google Translate**: Google Translate is one of the famous applications of Natural Language Processing. It helps convert written or spoken sentences into any language. Also, we can find the correct pronunciation and meaning of a word by using Google Translate. It uses advanced techniques of Natural Language Processing to achieve success in translating sentences into various languages.



2. **Chatbots**: To provide a better customer support service, companies have started using chatbots for 24/7 service. AI Chatbots help resolve the basic queries of customers. If a

chatbot is not able to resolve any query, then it forwards it to the support team, while still engaging the customer. It helps make customers feel that the customer support team is quickly attending to them. With the help of chatbots, companies have become capable of building cordial relations with customers. It is only possible with the help of Natural Language Processing.



3. What are stop words?

Stop words are said to be useless data for a search engine. Words such as articles, prepositions, etc. are considered stop words. There are stop words such as was, were, is, am, the, a, an, how, why, and many more. In Natural Language Processing, we eliminate the stop words to understand and analyze the meaning of a sentence. The removal of stop words is one of the most important tasks for search engines. Engineers design the algorithms of search engines in such a way that they ignore the use of stop words. This helps show the relevant search result for a query.

4. What is NLTK?

NLTK is a Python library, which stands for Natural Language Toolkit. We use NLTK to process data in human-spoken languages. NLTK allows us to apply techniques such as parsing, tokenization, lemmatization, stemming, and more to understand natural languages. It helps in categorizing text, parsing linguistic structure, analyzing documents, etc.

A few of the libraries of the NLTK package that we often use in NLP are:

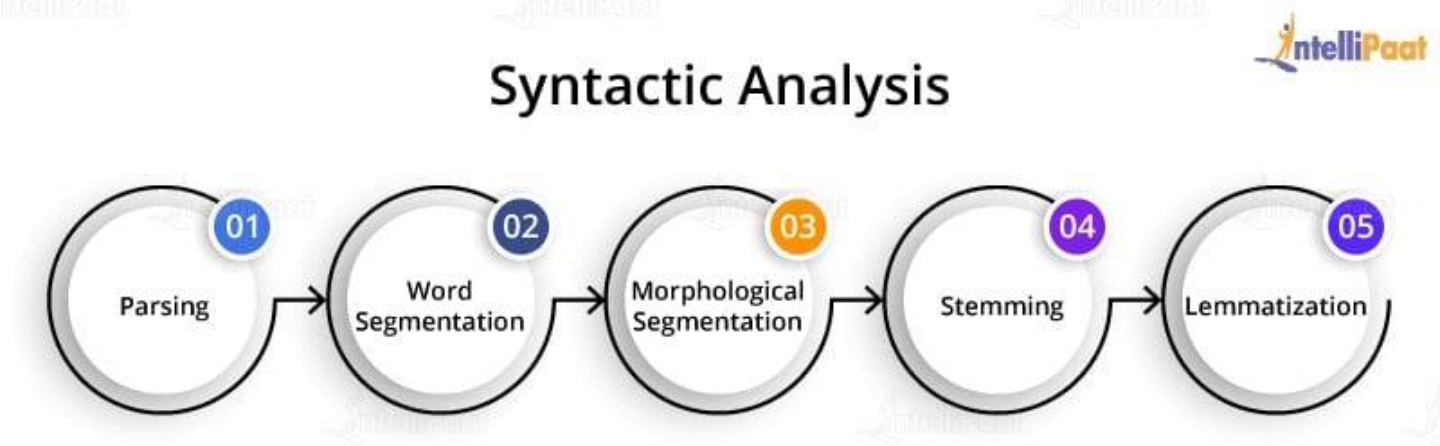
1. SequentialBackoffTagger
2. DefaultTagger
3. UnigramTagger

4. treebank
5. wordnet
6. FreqDist
7. patterns
8. RegexpTagger
9. backoff_tagger
10. UnigramTagger, BigramTagger, and TrigramTagger

5. What is Syntactic Analysis?

Syntactic analysis is a technique of analyzing sentences to extract meaning from them. Using syntactic analysis, a machine can analyze and understand the order of words arranged in a sentence. NLP employs grammar rules of a language that helps in the syntactic analysis of the combination and order of words in documents.

The techniques used for syntactic analysis are as follows:



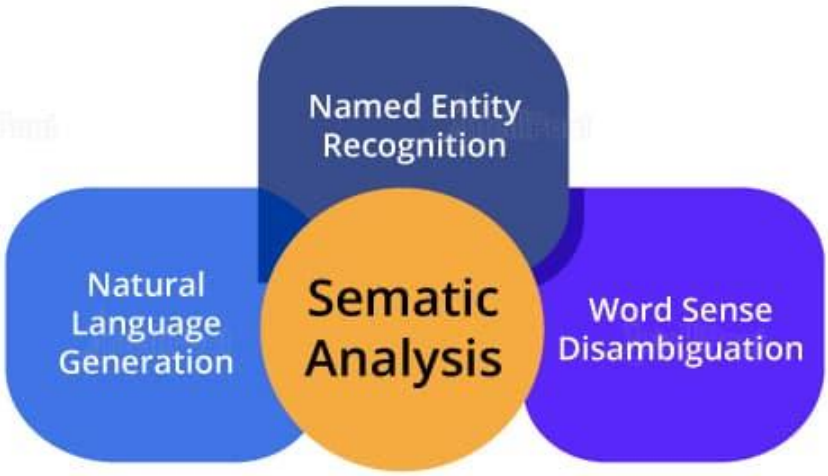
1. **Parsing**: It helps in deciding the structure of a sentence or text in a document. It helps analyze the words in the text based on the grammar of the language.
2. **Word segmentation**: The segmentation of words segregates the text into small significant units.
3. **Morphological segmentation**: The purpose of morphological segmentation is to break words into their base form.
4. **Stemming**: It is the process of removing the suffix from a word to obtain its root word.

5. **Lemmatization**: It helps combine words using suffixes, without altering the meaning of the word.

6. What is Semantic Analysis?

Semantic analysis helps make a machine understand the meaning of a text. It uses various algorithms for the interpretation of words in sentences. It also helps understand the structure of a sentence.

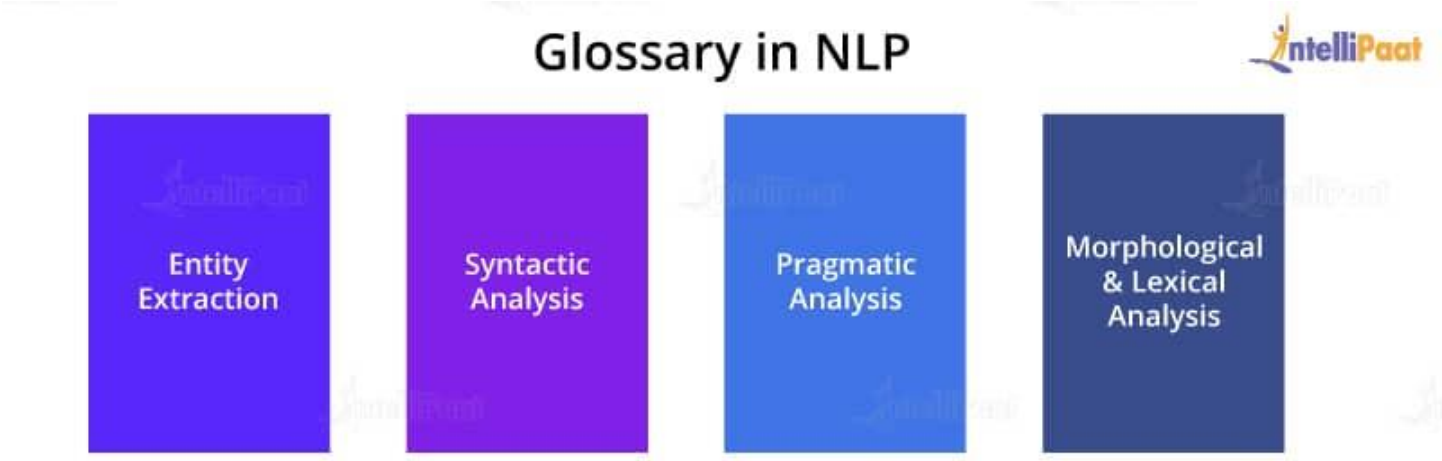
Techniques used for semantic analysis are as given below:



1. **Named entity recognition:** This is the process of information retrieval that helps identify entities such as the name of a person, organization, place, time, emotion, etc.
2. **Word sense disambiguation:** It helps identify the sense of a word used in different sentences.
3. **Natural language generation:** It is a process used by the software to convert structured data into human-spoken languages. By using NLG, organizations can automate content for custom reports.

7. List the components of Natural Language Processing.

The major components of NLP are as follows:



- **Entity extraction:** Entity extraction refers to the retrieval of information such as place, person, organization, etc. by the segmentation of a sentence. It helps in the recognition of an entity in a text.
- **Syntactic analysis:** Syntactic analysis helps draw the specific meaning of a text.
- **Pragmatic analysis:** To find useful information from a text, we implement pragmatic analysis techniques.
- **Morphological and lexical analysis:** It helps in explaining the structure of words by analyzing them through parsing.

8. What is Latent Semantic Indexing (LSI)?

Latent semantic indexing is a mathematical technique used to improve the accuracy of the information retrieval process. The design of LSI algorithms allows machines to detect the hidden (latent) correlation between semantics (words). To enhance information understanding, machines generate various concepts that associate with the words of a sentence.

The technique used for information understanding is called singular value decomposition. It is generally used to handle static and unstructured data. The matrix obtained for singular value decomposition contains rows for words and columns for documents. This method is best suited to identify components and group them according to their types.

The main principle behind LSI is that words carry a similar meaning when used in a similar context. Computational LSI models are slow in comparison to other models. However, they are good at contextual awareness which helps improve the analysis and understanding of a text or a document.

9. What are Regular Expressions?

A regular expression is used to match and tag words. It consists of a series of characters for matching strings.

Suppose, if A and B are regular expressions, then the following are true for them:

- If {ε} is a regular language, then ε is a regular expression for it.
- If A and B are regular expressions, then A + B is also a regular expression within the language {A, B}.
- If A and B are regular expressions, then the concatenation of A and B (A.B) is a regular expression.
- If A is a regular expression, then A* (A occurring multiple times) is also a regular expression.

10. What is Regular Grammar?

Regular grammar is used to represent a regular language.

Regular grammar comprises rules in the form of A -> a, A -> aB, and many more. The rules help detect and analyze strings by automated computation**.**

Regular grammar consists of four tuples:

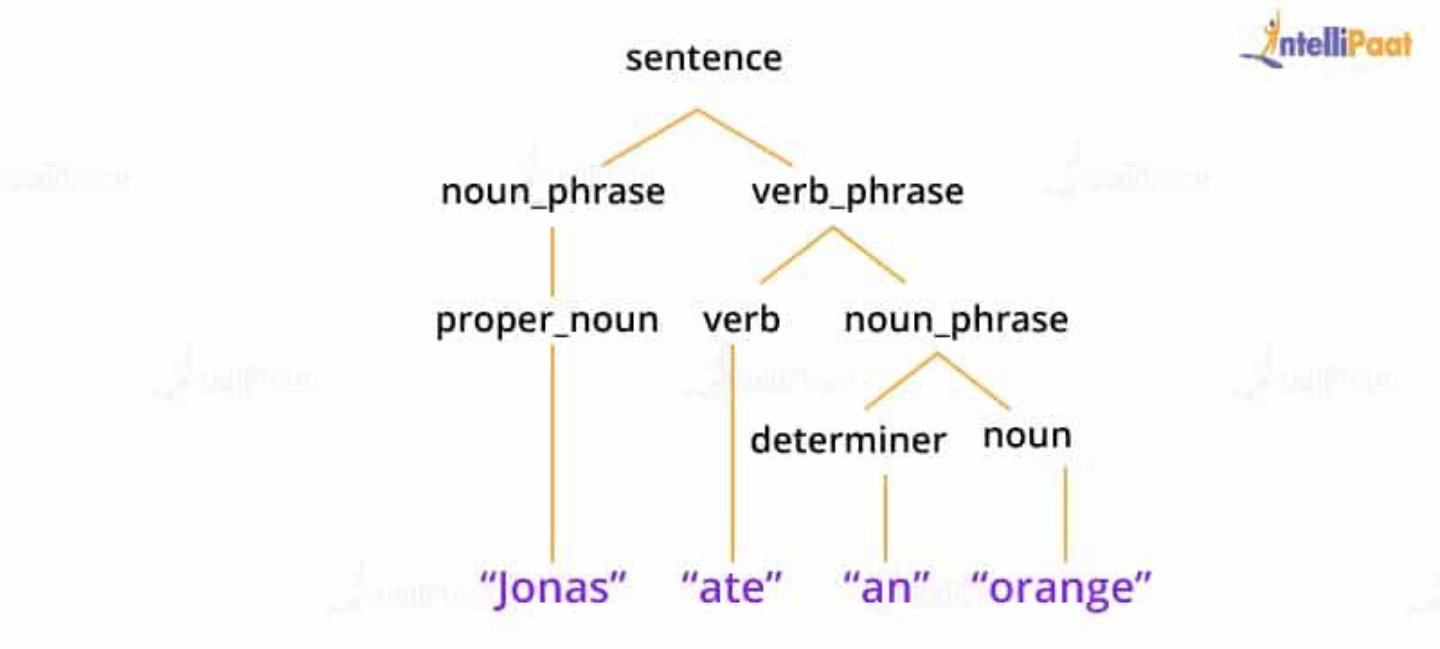
1. ‘N’ is used to represent the non-terminal set.
2. ‘∑’ represents the set of terminals.
3. ‘P’ stands for the set of productions.
4. ‘S € N’ denotes the start of non-terminal.

Regular grammar is of 2 types:
(a) Left Linear Grammar(LLG)

(b) Right Linear Grammar(RLG)

11. What is Parsing in the context of NLP?

Parsing in NLP refers to the understanding of a sentence and its grammatical structure by a machine. Parsing allows the machine to understand the meaning of a word in a sentence and the grouping of words, phrases, nouns, subjects, and objects in a sentence. Parsing helps analyze the text or the document to extract useful insights from it. To understand parsing, refer to the below diagram:



In this, 'Jonas ate an orange' is parsed to understand the structure of the sentence.

Intermediate NLP Interview Questions

12. What is TF-IDF?

TFIDF or Term Frequency-Inverse Document Frequency indicates the importance of a word in a set. It helps in information retrieval with numerical statistics. For a specific document, TF-IDF shows a frequency that helps identify the keywords in a document. The major use of TF-IDF in NLP is the extraction of useful information from crucial documents by statistical data. It is ideally used to classify and summarize the text in documents and filter out stop words.

TF helps calculate the ratio of the frequency of a term in a document and the total number of terms. Whereas, **IDF** denotes the importance of the term in a document.

The formula for calculating TF-IDF:

TF(W) = (Frequency of W in a document)/(The total number of terms in the document)

IDF(W) = log_e(The total number of documents/The number of documents having the term W)

When **TF*IDF** is high, the frequency of the term is less and vice versa.

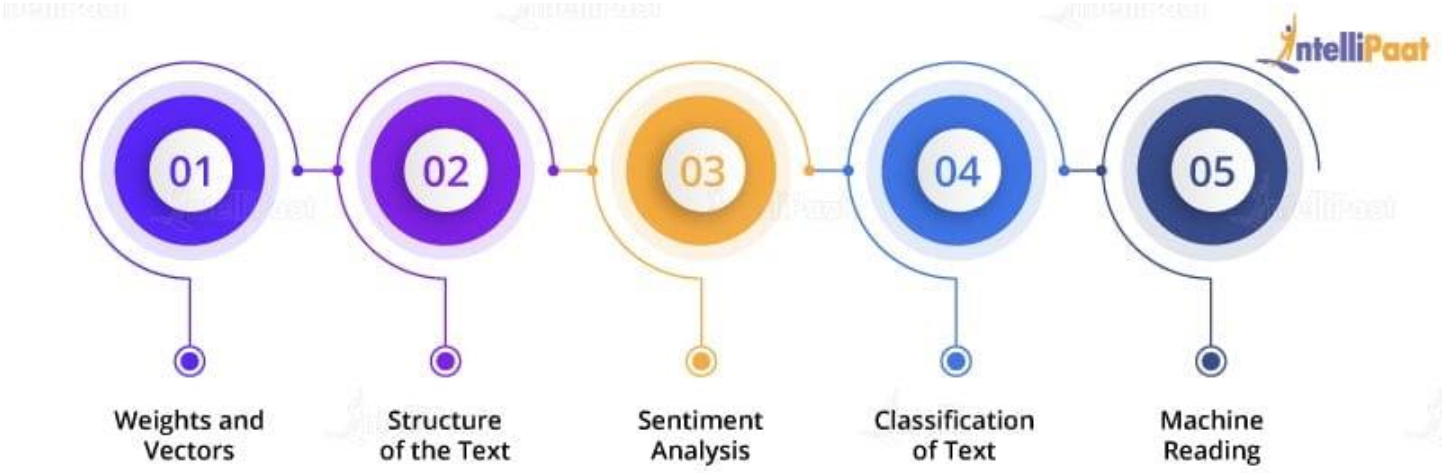
Google uses TF-IDF to decide the index of search results according to the relevancy of pages. The design of the TF-IDF algorithm helps optimize the search results in Google. It helps quality content rank up in search results.

If you want to know more about 'What is Natural Language Processing?' you can go through this Natural Language Processing Using Python course!

13. Define the terminology in NLP.

This is one of the most often asked NLP interview questions.

The interpretation of Natural Language Processing depends on various factors, and they are:



Weights and Vectors

- Use of TF-IDF for information retrieval
- Length (TF-IDF and doc)
- Google Word Vectors
- Word Vectors

Structure of the Text

- POS tagging
- Head of the sentence
- Named Entity Recognition (NER)

Sentiment Analysis

- Knowledge of the characteristics of sentiment
- Knowledge about entities and the common dictionary available for sentiment analysis

Classification of Text

- Supervised learning algorithm
- Training set
- Validation set
- Test set
- Features of the text
- LDA

Machine Reading

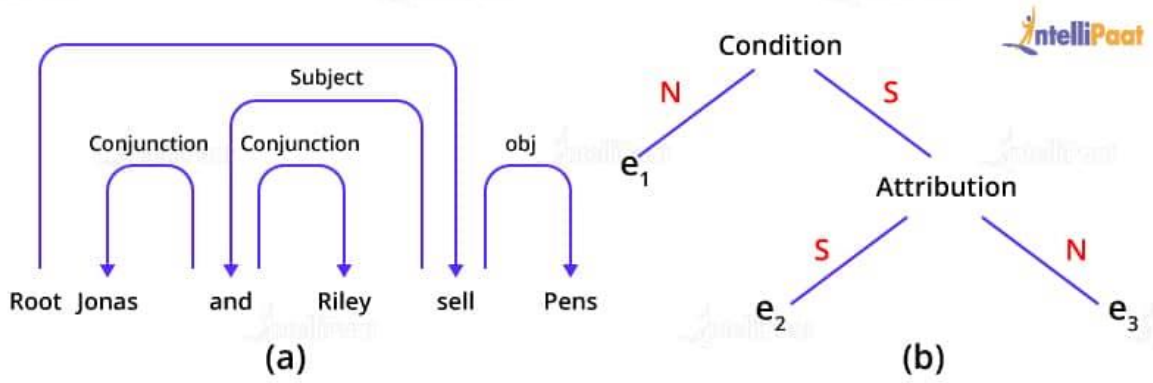
- Removal of possible entities
- Joining with other entities
- DBpedia

14. Explain Dependency Parsing in NLP.

Dependency parsing helps assign a syntactic structure to a sentence. Therefore, it is also called syntactic parsing. Dependency parsing is one of the critical tasks in NLP. It allows the analysis of a sentence using parsing algorithms. Also, by using the parse tree in dependency parsing, we can check the grammar and analyze the semantic structure of a sentence.

For implementing dependency parsing, we use the spaCy package. It implements token properties to operate the dependency parse tree.

The below diagram shows the dependency parse tree:



15. What is the difference between NLP and NLU?

The below table shows the difference between NLP and NLU:

Natural Language Processing (NLP)	Natural Language Understanding (NLU)
Used to create systems capable of establishing communication between humans and computers	Provides techniques to solve complicated problems related to machine understanding
Takes care of all the techniques required for the interaction between machines and humans	Helps convert the unorganized input data into a structured format to allow the machine to understand the data
Provides techniques to analyze 'What is said?'	Helps understand 'What is meant?'

16. What is the difference between NLP and CI?

The below table shows the difference between NLP and C

Natural Language Processing (NLP)	Conversational Interfaces (CI)
Helps analyze and identify user requests	A conversational interface that helps mix images, videos, audios with menus, buttons, etc.
Helps understand and interpret everything about a text or a document	Provides only the necessary information to users
Used by chatbots for providing customer support	Helps provide better customer service by understanding user queries and responding to them, providing a personalized experience

17. What is Pragmatic Analysis?

Pragmatic analysis is an important task in NLP for interpreting knowledge that is lying outside a given document. The aim of implementing pragmatic analysis is to focus on exploring a different aspect of the document or text in a language. This requires a comprehensive knowledge of the real world. The pragmatic analysis allows software applications for the critical interpretation of the real-world data to know the actual meaning of sentences and words.

Example:

Consider this sentence: ‘Do you know what time it is?’

This sentence can either be asked for knowing the time or for yelling at someone to make them note the time. This depends on the context in which we use the sentence.

18. What is Pragmatic Ambiguity?

Pragmatic ambiguity refers to the multiple descriptions of a word or a sentence. An ambiguity arises when the meaning of the sentence is not clear. The words of the sentence may have different meanings. Therefore, in practical situations, it becomes a challenging task for a machine to understand the meaning of a sentence. This leads to pragmatic ambiguity.

Example:

Check out the below sentence.

‘Are you feeling hungry?’

The given sentence could be either a question or a formal way of offering food.

19. What are unigrams, bigrams, trigrams, and n-grams in NLP?

When we parse a sentence one word at a time, then it is called a unigram. The sentence parsed two words at a time is a bigram.

When the sentence is parsed three words at a time, then it is a trigram. Similarly, n-gram refers to the parsing of *n* words at a time.

Example: To understand unigrams, bigrams, and trigrams, you can refer to the below diagram:

Therefore, parsing allows machines to understand the individual meaning of a word in a sentence. Also, this type of parsing helps predict the next word and correct spelling errors.

20. What are the steps involved in solving an NLP problem?

Below are the steps involved in solving an NLP problem:

1. Gather the text from the available dataset or by web scraping
2. Apply stemming and lemmatization for text cleaning
3. Apply feature engineering techniques
4. Embed using **word2vec**
5. Train the built model using neural networks or other Machine Learning techniques
6. Evaluate the model's performance
7. Make appropriate changes in the model
8. Deploy the model

21. What is Feature Extraction in NLP?

Features or characteristics of a word help in text or document analysis. They also help in sentiment analysis of a text. Feature extraction is one of the techniques that are used by recommendation systems. Reviews such as 'excellent,' 'good,' or 'great' for a movie are positive reviews, recognized by a recommender system. The recommender system also tries to identify the features of the text that help in describing the context of a word or a sentence. Then, it makes a group or category of the

words that have some common characteristics. Now, whenever a new word arrives, the system categorizes it as per the labels of such groups.

22. What are precision and recall?

The metrics used to test an NLP model are precision, recall, and F1. Also, we use accuracy for evaluating the model's performance. The ratio of prediction and the desired output yields the accuracy of the model.

Precision is the ratio of true positive instances and the total number of positively predicted instances.

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$= \frac{True\ Positive}{Total\ Predicted\ Positive}$$

Recall is the ratio of true positive instances and the total actual positive instances.

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive}$$

23. What is F1 score in NLP?

F1 score evaluates the weighted average of recall and precision. It considers both false negative and false positive instances while evaluating the model. F1 score is more accountable than accuracy for an NLP model when there is an uneven distribution of class. Let us look at the formula for calculating F1 score:

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Advanced NLP Interview Questions

24. How to tokenize a sentence using the nltk package?

Tokenization is a process used in NLP to split a sentence into tokens. **Sentence tokenization** refers to splitting a text or paragraph into sentences.

For tokenizing, we will import **sent_tokenize** from the **nltk package**:

```
from nltk.tokenize import sent_tokenize
```

We will use the below paragraph for sentence tokenization:

Para = “Hi Guys. Welcome to Intellipaat. This is a blog on the NLP interview questions and answers.”

```
sent_tokenize(Para)
```

Output:

```
[ 'Hi Guys.','Welcome to Intellipaat. ', 'This is a blog on the NLP interview questions and answers.']
```

Tokenizing a word refers to splitting a sentence into words.

Now, to tokenize a word, we will import **word_tokenize** from the nltk package.

```
from nltk.tokenize import word_tokenize
```

Para = “Hi Guys. Welcome to Intellipaat. This is a blog on the NLP interview questions and answers.”

```
word_tokenize(Para)
```

Output:

```
[ 'Hi', 'Guys', ' . ', 'Welcome' , 'to' , 'Intellipaat' , ' . ' , 'This' , 'is' ,  'a', 'blog' , 'on' , 'the' , 'NLP' , 'interview' , 'questions' , 'and' , 'answers' , ' . ' ]
```

25. Explain how we can do parsing.

Parsing is the method to identify and understand the syntactic structure of a text. It is done by analyzing the individual elements of the text. The machine parses the text one word at a time, then two at a time, further three, and so on.

- When the machine parses the text one word at a time, then it is a **unigram**.
- When the text is parsed two words at a time, it is a **bigram**.
- The set of words is a **trigram** when the machine parses three words at a time.

Look at the below diagram to understand unigram, bigram, and trigram.

Now, let's implement parsing with the help of the **nltk** package.

```
import nltk
text = ”Top 30 NLP interview questions and answers”
```

We will now tokenize the text using **word_tokenize**.

```
text_token= word_tokenize(text)
```

Now, we will use the function for extracting unigrams, bigrams, and trigrams.

```
list(nltk.unigrams(text))
```

Output:

```
[ "Top 30 NLP interview questions and answer"]
list(nltk.bigrams(text))
```

Output:

```
["Top 30", "30 NLP", "NLP interview", "interview questions",  "questions and", "and answer"]
list(nltk.trigrams(text))
```

Output:

```
["Top 30 NLP", "NLP interview questions", "questions and answers"]
```

For extracting **n-grams**, we can use the function **nltk.ngrams** and give the argument *n* for the number of parsers.

```
list(nltk.ngrams(text,n))
```

26. Explain Stemming with the help of an example.

In Natural Language Processing, stemming is the method to extract the root word by removing suffixes and prefixes from a word.

For example, we can reduce ‘stemming’ to ‘stem’ by removing ‘m’ and ‘ing.’

We use various algorithms for implementing stemming, and one of them is PorterStemmer.

First, we will import **PorterStemmer** from the nltk package.

```
from nltk.stem import PorterStemmer
```

Creating an object for PorterStemmer

```
pst=PorterStemmer()
pst.stem(“running”), pst.stem(“cookies”), pst.stem(“flying”)
```

Output:

```
(‘run’, ‘cooki', ‘fly’ )
```

27. Explain Lemmatization with the help of an example.

We use stemming and lemmatization to extract root words. However, stemming may not give the actual word, whereas lemmatization generates a meaningful word.

In lemmatization, rather than just removing the suffix and the prefix, the process tries to find out the root word with its proper meaning.

Example: ‘Bricks’ becomes ‘brick,’ ‘corpora’ becomes ‘corpus,’ etc.

Let’s implement lemmatization with the help of some nltk packages.

First, we will import the required packages.

```
from nltk.stem import wordnet
from nltk.stem import WordnetLemmatizer
```

Creating an object for WordnetLemmatizer()

```
lemma= WordnetLemmatizer()
list = [“Dogs”, “Corpora”, “Studies”]
for n in list:
print(n + “:” + lemma.lemmatize(n))
```

Output:

```
Dogs: Dog
Corpora: Corpus
Studies: Study
```

28. What is Parts-of-speech Tagging?

The parts-of-speech (POS) tagging is used to assign tags to words such as nouns, adjectives, verbs, and more. The software uses the POS tagging to first read the text and then differentiate the words by tagging. The software uses algorithms for the parts-of-speech tagging. POS tagging is one of the most essential tools in Natural Language Processing. It helps in making the machine understand the meaning of a sentence.

We will look at the implementation of the POS tagging using stop words.

Let's import the required nltk packages.

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize, sent_tokenize
stop_words = set(stopwords.words('english'))
txt = "Sourav, Pratyush, and Abhinav are good friends."
```

Tokenizing using sent_tokenize

```
tokenized_text = sent_tokenize(txt)
```

To find punctuation and words in a string, we will use **word_tokenizer** and then remove the stop words.

```
for n in tokenized_text:
wordsList = nltk.word_tokenize(i)
wordsList = [w for w in wordsList if not w instop_words]
```

Now, we will use the POS tagger.

```
tagged_words = nltk.pos_tag(wordsList)
print(tagged_words)
```

Output:

```
[('Sourav', 'NNP'), ('Pratyush', 'NNP'), ('Abhinav', 'NNP'), ('good',  'JJ'), ('friends', 'NNS')]
```

29. Explain Named Entity Recognition by implementing it.

Named Entity Recognition (NER) is an information retrieval process. NER helps classify named entities such as monetary figures, location, things, people, time, and more. It allows the software to analyze and understand the meaning of the text. NER is mostly used in NLP, Artificial Intelligence, and Machine Learning. One of the real-life applications of NER is chatbots used for customer support.

Let’s implement NER using the spaCy package.

Importing the spaCy package:

```
import spacy
nlp = spacy.load('en_core_web_sm')
Text = "The head office of Google is in California"
document = nlp(text)
for ent in document.ents:
print(ent.text, ent.start_char, ent.end_char, ent.label_)
```

Output:

```
Office 9 15 Place
Google 19 25 ORG
California 32 41 GPE
```

Note: Office 9 15 Place means word starts at 9th position when tokenized and ends at 15, this is inclusive of spaces.

30. How to check word similarity using the spaCy package

To find out the similarity among words, we use word similarity. We evaluate the similarity with the help of a number that lies between 0 and 1. We use the spacy library to implement the technique of word similarity.

```
import spacy
nlp = spacy.load('en_core_web_md')
print("Enter the words")
input_words = input()
tokens = nlp(input_words)
for i in tokens:
print(i.text, i.has_vector, i.vector_norm, i.is_oov)
token_1, token_2 = tokens[0], tokens[1]
print("Similarity between words:", token_1.similarity(token_2))
```

Output:

```
hot  True 5.6898586 False
cold True6.5396233 False
Similarity: 0.597265
```

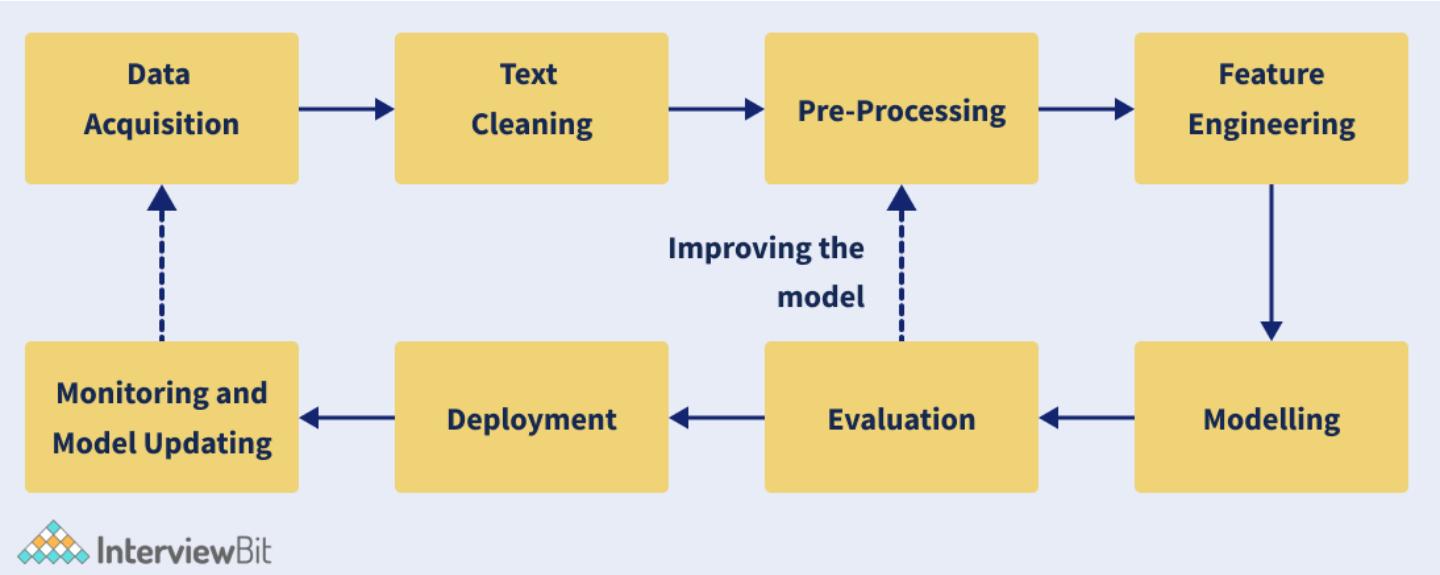
INTERVIEWBIT

(https://www.interviewbit.com/nlp-interview-questions/)

NLP Interview Questions for Freshers

1. What are the stages in the lifecycle of a natural language processing (NLP) project?

Following are the stages in the lifecycle of a natural language processing (NLP) project:

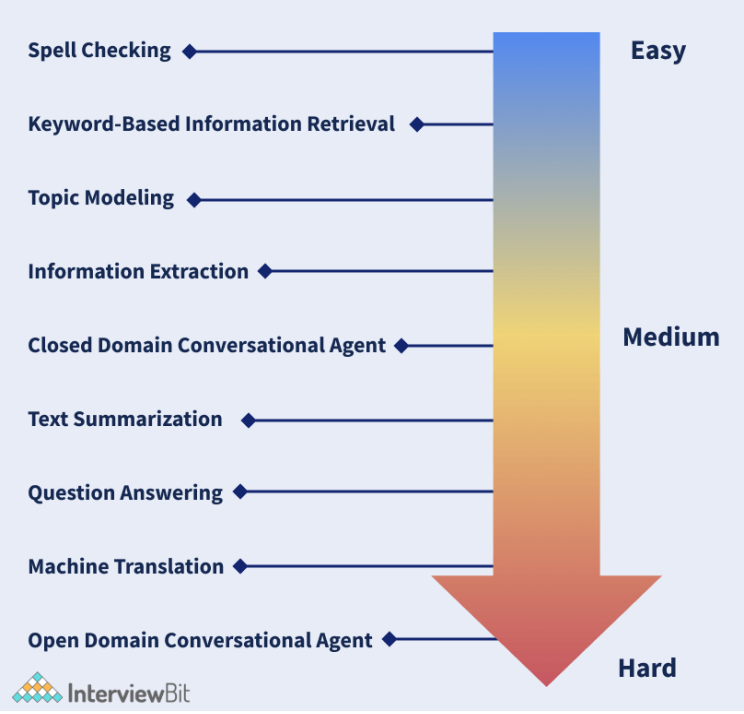


- **Data Collection:** The procedure of collecting, measuring, and evaluating correct insights for research using established approved procedures is referred to as data collection.
- **Data Cleaning:** The practice of correcting or deleting incorrect, corrupted, improperly formatted, duplicate, or incomplete data from a dataset is known as data cleaning.
- **Data Pre-Processing:** The process of converting raw data into a comprehensible format is known as data preparation.
- **Feature Engineering:** Feature engineering is the process of extracting features (characteristics, qualities, and attributes) from raw data using domain expertise.
- **Data Modeling:** The practice of examining data objects and their relationships with other things is known as data modelling. It's utilised to look into the data requirements for various business activities.
- **Model Evaluation:** Model evaluation is an important step in the creation of a model. It aids in the selection of the best model to represent our data and the prediction of how well the chosen model will perform in the future.
- **Model Deployment:** The technical task of exposing an ML model to real-world use is known as model deployment.
- **Monitoring and Updating:** The activity of measuring and analysing production model performance to ensure acceptable quality as defined by the use case is known as machine learning monitoring. It delivers alerts about performance difficulties and assists in diagnosing and resolving the core cause.

2. What are some of the common NLP tasks?

Some of the common tasks of NLP include:

- **Machine Translation:** This helps in translating a given piece of text from one language to another.
- **Text Summarization:** Based on a large corpus, this is used to give a short summary that gives an idea of the entire text in the document.
- **Language Modeling:** Based on the history of previous words, this helps uncover what the further sentence will look like. A good example of this is the auto-complete sentences feature in Gmail.
- **Topic Modelling:** This helps uncover the topical structure of a large collection of documents. This indicates what topic a piece of text is actually about.
- **Question Answering:** This helps prepare answers automatically based on a corpus of text, and on a question that is posed.
- **Conversational Agent:** These are basically voice assistants that we commonly see such as Alexa, Siri, Google Assistant, Cortana, etc.
- **Information Retrieval:** This helps in fetching relevant documents based on a user's search query.
- **Information Extraction:** This is the task of extracting relevant pieces of information from a given text, such as calendar events from emails.
- **Text Classification:** This is used to create a bucket of categories of a given text, based on its content. This is used in a wide variety of AI-based applications such as sentiment analysis and spam detection.



Common NLP Tasks in order of Difficulty

3. What are the different approaches used to solve NLP problems?

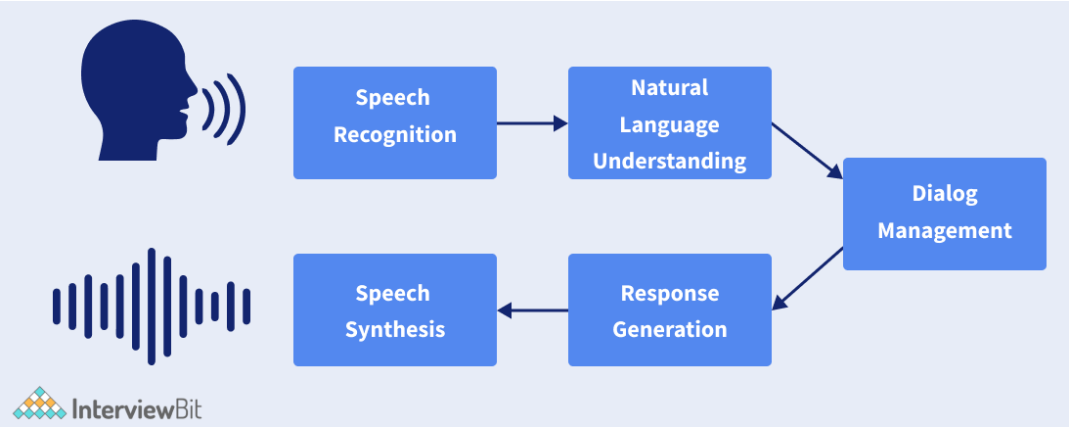
There are multiple approaches to solving NLP problems. These usually come in 3 categories:

- Heuristics
- Machine learning
- Deep Learning

4. How do Conversational Agents work?

The following NLP components are used in Conversational Agents:

- **Speech Recognition and Synthesis:** In the first stage, speech recognition helps convert speech signals to their phonemes, and are then transcribed as words.
- **Natural Language Understanding (NLU):** Here, the transcribed text from stage one is further analysed through AI techniques within the natural language understanding system. Certain NLP tasks such as Named Entity Recognition, Text Classification, Language modelling, etc. come into play here.
- **Dialog Management:** Once the needed information from text is extracted, we move on to the stage of understanding the user's intent. The user's response can then be classified by using a text classification system as a pre-defined intent. This helps the conversational agent in figuring out what is actually being asked.
- **Generating Response:** Based on the above stages, the agent generates an appropriate response that is based on a semantic interpretation of the user's intent.



5. What is meant by data augmentation? What are some of the ways in which data augmentation can be done in NLP projects?

NLP has some methods through which we can take a small dataset and use that in order to create more data. This is called data augmentation. In this, we use language properties to create text that is syntactically similar to the source text data.

Some of the ways in which data augmentation can be done in NLP projects are as follows:

- Replacing entities
- TF-IDF–based word replacement
- Adding noise to data

- Back translation
- Synonym replacement
- Bigram flipping

6. How can data be obtained for NLP projects?

There are multiple ways in which data can be obtained for NLP projects. Some of them are as follows:

- **Using publicly available datasets:** Datasets for NLP purposes are available on websites like Kaggle as well as Google Datasets.
- **By using data augmentation:** These are used to create additional datasets from existing datasets.
- **Scraping data from the web:** Using coding in Python or other languages once can scrape data from websites that are usually not readily available in a structured form.

7. What do you mean by Text Extraction and Cleanup?

The process of extracting raw text from the input data by getting rid of all the other non-textual information, such as markup, metadata, etc., and converting the text to the required encoding format is called **text extraction and cleanup**. Usually, this depends on the format of available data for the required project.

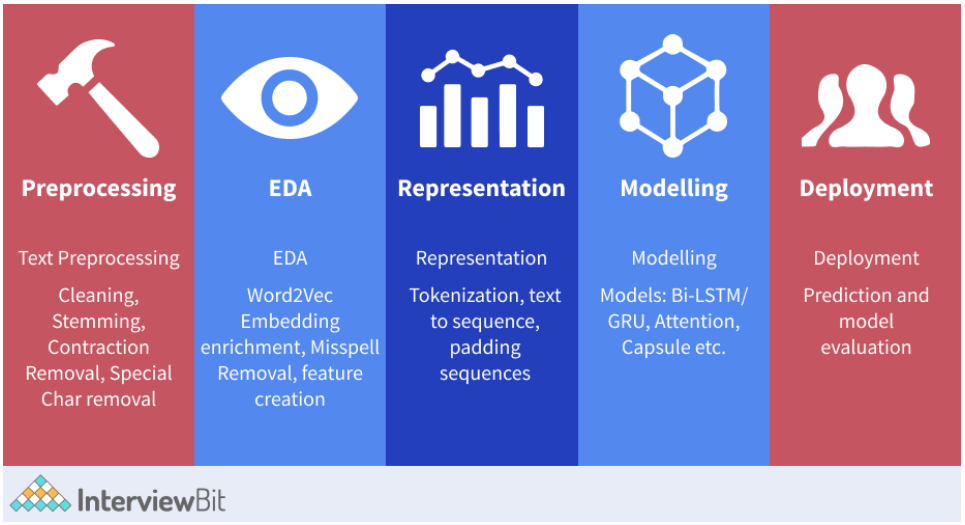
Following are the common ways used for Text Extraction in NLP:

- Named Entity Recognition
- Sentiment Analysis
- Text Summarization
- Aspect Mining
- Topic Modeling

8. What are the steps involved in preprocessing data for NLP?

Here are some common pre-processing steps used in NLP software:

- **Preliminaries:** This includes word tokenization and sentence segmentation.
- **Common Steps:** Stop word removal, stemming and lemmatization, removing digits/punctuation, lowercasing, etc.
- **Processing Steps:** Code mixing, normalization, language detection, transliteration, etc.
- **Advanced Processing:** Parts of Speech (POS) tagging, coreference resolution, parsing, etc.



9. What do you mean by Stemming in NLP?

When we remove the suffixes from a word so that the word is reduced to its base form, this process is called stemming. When the word is reduced to its base form, all the different variants of that word can be represented by the same form (e.g., “bird” and “birds” are both reduced to “bird”).

We can do this by using a fixed set of rules. For instance: if a word ends in “-es,” we can remove the “-es”).

Even though these rules might not really make sense as a linguistically correct base form, stemming is usually carried out to match user queries in search engines to relevant documents. And in text classification, is done to reduce the feature space to train our machine learning (ML) models.

The code snippet given below depicts the way to use a well known NLP algorithm for stemming called Porter Stemmer using NLTK:

```
from nltk.stem.porter import PorterStemmer
stemmer = PorterStemmer()
word1, word2 = "bikes", "revolution"
print(stemmer.stem(word1), stemmer.stem(word2))
```

This gives “bike” as the stemmed version for “bikes,” but “revolut” as the stemmed form of “revolution,” even though the latter is not linguistically correct. Even if this might not affect the performance of the search engine, a derivation of the correct linguistic form becomes useful in some other cases. This can be done by another process that is closer to stemming, known as lemmatization.

10. What do you mean by Lemmatization in NLP?

The method of mapping all the various forms of a word to its base word (also called “lemma”) is known as Lemmatization. Although this may appear close to the definition of stemming, these are actually different. For instance, the word “better,” after stemming, remains the same. However, upon lemmatization, this should become “good,”. Lemmatization needs greater linguistic knowledge. Modelling and developing efficient lemmatizers still remains an open problem in NLP research.

The application of a lemmatizer based on WordNet from NLTK is shown in the code snippet below:

```
from nltk.stem import WordNetLemmatizer
lemmatizer = WordnetLemmatizer()
print(lemmatizer.lemmatize("better", pos="a")) #a is for adjective
```

NLP Interview Questions for Experienced

11. What is the meaning of Text Normalization in NLP?

Consider a situation in which we're operating with a set of social media posts to find information events. Social media textual content may be very exceptional from the language we'd see in, say, newspapers. A phrase may be spelt in multiple ways, such as in shortened forms, (for instance, with and without hyphens), names are usually in lowercase, and so on. When we're developing NLP tools to work with such kinds of data, it's beneficial to attain a canonical representation of textual content that captures these kinds of variations into one representation. This is referred to as text normalization.

Converting all text to lowercase or uppercase, converting digits to text (e.g., 7 to seven), expanding abbreviations, and so on are some frequent text normalisation stages.

12. Explain the concept of Feature Engineering.

After a variety of pre-processing procedures and their applications, we need a way to input the pre-processed text into an NLP algorithm later when we employ ML methods to complete our modelling step. The set of strategies that will achieve this goal is referred to as feature engineering. Feature extraction is another name for it. The purpose of feature engineering is to convert the text's qualities into a numeric vector that NLP algorithms can understand. This stage is called "text representation".

13. What is an ensemble method in NLP?

An ensemble approach is a methodology that derives an output or makes predictions by combining numerous independent similar or distinct models/weak learners. An ensemble can also be created by combining various models such as random forest, SVM, and logistic regression.

Bias, variance, and noise, as we all know, have a negative impact on the mistakes and predictions of any machine learning model. Ensemble approaches are employed to overcome these drawbacks.

14. What do you mean by TF-IDF in Natural language Processing?

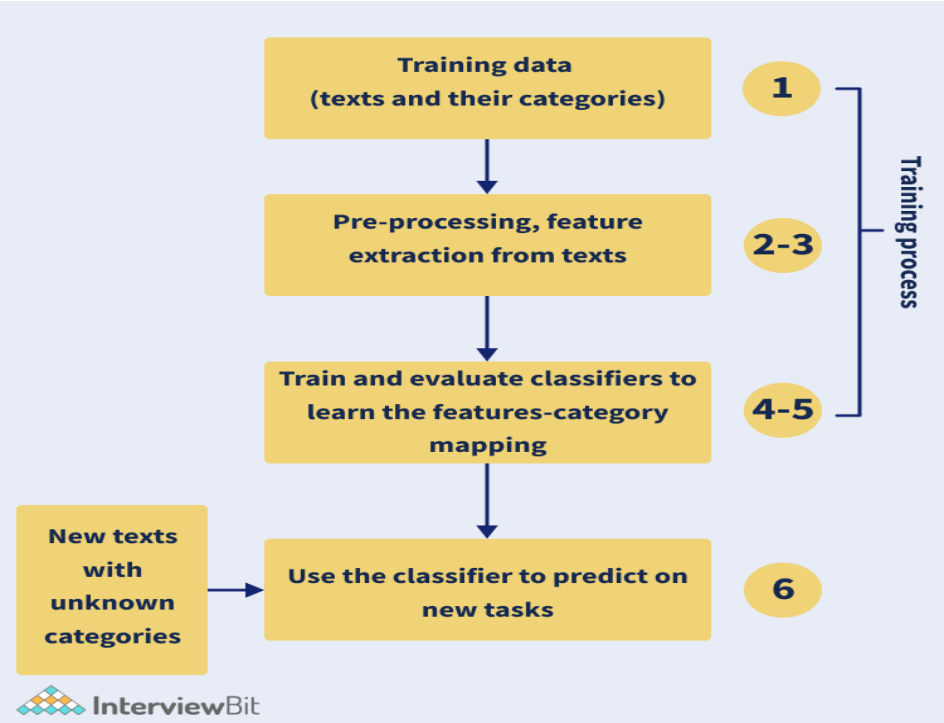
TF-IDF also called **Term Frequency-Inverse Document Frequency** helps us get the importance of a particular word relative to other words in the corpus. It's a common scoring metric in information retrieval (IR) and summarization. TF-IDF converts words into vectors and adds semantic information, resulting in weighted unusual words that may be utilised in a variety of NLP applications.

15. What are the steps to follow when building a text classification system?

When creating a text classification system, the following steps are usually followed:

- Gather or develop a labelled dataset that is appropriate for the purpose.

- Decide on an evaluation metric after splitting the dataset into two (training and test) or three parts: training, validation (i.e., development), and test sets (s).
- Convert unprocessed text into feature vectors.
- Utilize the feature vectors and labels from the training set to train a classifier.
- Benchmark the model's performance on the test set using the evaluation metric(s) from Step 2.
- Deploy the model and track its performance to serve a real-world use case.



16. Explain how parsing is done in NLP.

Parsing is the process of identifying and understanding a text's syntactic structure. It is accomplished by examining the text's constituent pieces. The machine parses each word one by one, then two by two, three by three, and so on. It's a unigram when the system parses the text one word at a time. A bigram is a text that is parsed two words at a time. When the machine parses three words at a time, the set of words is called a **trigram**.

The following points will help us comprehend the importance of parsing in NLP:

- Any syntax errors are reported by the parser.
- It aids in the recovery of often occurring errors so that the remainder of the programme can be processed.
- A parser is used to generate the parse tree.
- The parser is used to construct a symbol table, which is crucial in NLP.
- In addition, a Parser is utilised to generate intermediate representations (IR).

17. What do you mean by a Bag of Words (BOW)?

The **Bag of Words** model is a popular one that uses word frequency or occurrences to train a classifier. This methodology generates a matrix of occurrences for documents or phrases, regardless of their grammatical structure or word order.

A bag-of-words is a text representation that describes the frequency with which words appear in a document. It entails two steps:

- A list of terms that are well-known.
- A metric for determining the existence of well-known terms.

Because any information about the sequence or structure of words in the document is deleted, it is referred to as a "bag" of words. The model simply cares about whether or not recognised terms appear in the document, not where they appear.

18. What do you mean by Parts of Speech (POS) tagging in NLP?

A Part-Of-Speech Tagger (POS Tagger) reads the text in a language and assigns parts of speech to each word (and other tokens), such as noun, verb, adjective, and so on.

To label terms in text bodies, PoS taggers employ an algorithm. With tags like "noun-plural" or even more complicated labels, these taggers create more complex categories than those stated as basic PoS.

19. What is Latent Semantic Indexing (LSI) in NLP?

Latent Semantic Indexing (LSI), also known as Latent Semantic Analysis, is a mathematical method for improving the accuracy of information retrieval. It aids in the discovery of hidden(latent) relationships between words (semantics) by generating a set of various concepts associated with the terms of a phrase in order to increase information comprehension. Singular value decomposition is the NLP technique utilised for this aim. It's best for working with small groups of static documents.

20. What is the difference between NLP and NLU?

Natural Language Processing (NLP)	Natural Language Understanding (NLU)
NLP is a system that manages end-to-end conversations between computers and people at the same time.	NLU aids in the solving of Artificial Intelligence's complex problems.
Humans and machines are both involved in NLP.	NLU allows machines to interpret unstructured inputs by transforming them into structured text.
NLP focuses on interpreting language in its most literal sense, such as what was said.	NLU, on the other hand, concentrates on extracting context and meaning, or what was meant.
NLP can parse text-based on grammar, structure, typography, and point of view.	It'll be NLU that helps the machine deduce the meaning behind the language content.

21. What are some metrics on which NLP models are evaluated?

The following are some metrics on which NLP models are evaluated:

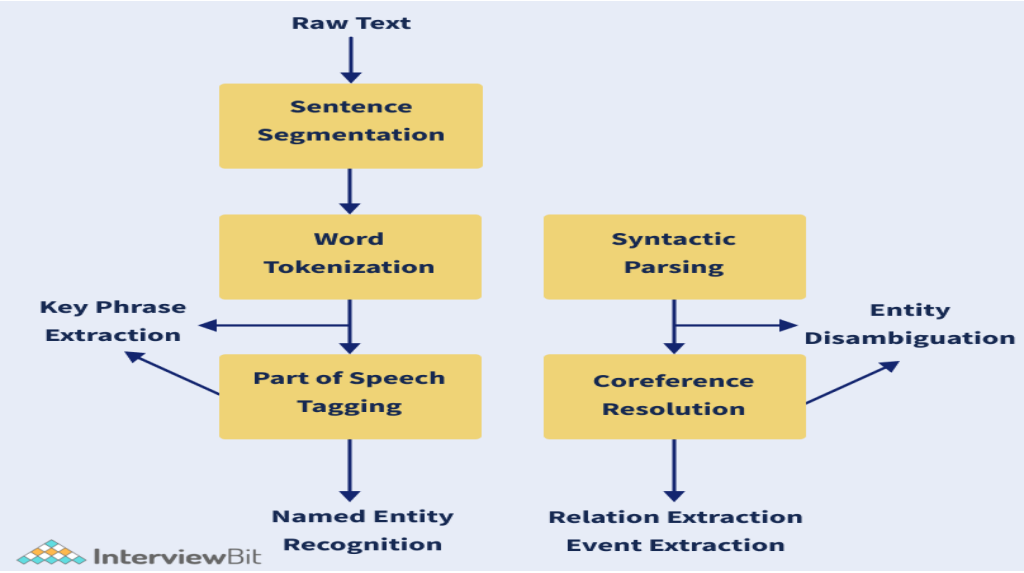
- **Accuracy:** When the output variable is categorical or discrete, accuracy is used. It is the percentage of correct predictions made by the model compared to the total number of predictions made.

- **Precision:** Indicates how precise or exact the model's predictions are, i.e., how many positive (the class we care about) examples can the model correctly identify given all of them?
- **Recall:** Precision and recall are complementary. It measures how effectively the model can recall the positive class, i.e., how many of the positive predictions it generates are correct.
- **F1 score:** This metric combines precision and recall into a single metric that also represents the trade-off between accuracy and recall, i.e., completeness and exactness.
  (2 Precision Recall) / (Precision + Recall) is the formula for F1.
- **AUC:** As the prediction threshold is changed, the AUC captures the number of correct positive predictions versus the number of incorrect positive predictions.

22. Explain the pipeline for Information extraction (IE) in NLP.

In comparison to text classification, the typical pipeline for IE necessitates more fine-grained NLP processing. For example, we'd need to know the part-of-speech tags of words to identify named entities (people, organisations, etc.). We would require coreference resolution to connect various references to the same entity (e.g., Albert Einstein, Einstein, the scientist, he, etc.). It's worth noting that none of these stages are required for creating a text classification system. As a result, IE is a more NLP-intensive operation than text categorization. Not all steps in the pipeline are required for all IE jobs, as shown in the diagram, and the figure shows which IE tasks necessitate which degrees of analysis.

Other than named entity recognition, all other IE tasks require deeper NLP pre-processing followed by models developed for those specific tasks. Key phrase extraction is the task that requires the least amount of NLP processing (some algorithms also do POS tagging before extracting key phrases), whereas all other IE tasks require deeper NLP pre-processing followed by models developed for those specific tasks. Standard evaluation sets are often used to assess IE tasks in terms of precision, recall, and F1 scores. Because of the various levels of NLP pre-processing required, the accuracy of these processing steps has an impact on IE jobs. All of these factors should be considered when collecting relevant training data and, if necessary, training our own models for IE.

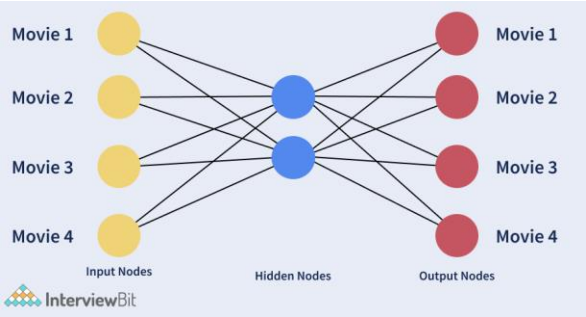


23. What do you mean by Autoencoders?

A network that is used for learning a vector representation of the input in a compressed form, is called an autoencoder. It is a type of unsupervised learning since labels aren't needed for the process. This is mainly used to learn the mapping function from the input. In order to make the mapping useful, the input is reconstructed from the vector representation. After training is complete, the vector representation that we get helps encode the input text as a dense vector. Autoencoders are generally used to make feature representations.

In the figure below, the hidden layer depicts a compressed representation of the source data that captures its essence. The input representation is reconstructed by the output layer called the

Decoder



24. What do you mean by Masked language modelling?

Masked language modelling is an NLP technique for extracting the output from a contaminated input. Learners can use this approach to master deep representations in downstream tasks. Using this NLP technique, you may predict a word based on the other words in the sentence.

The following is the process for Masked language modelling:

- Our text is tokenized. We start with text tokenization, just as we would with transformers.
- Make a tensor of labels. We're using a labels tensor to calculate loss against — and optimise towards — as we train our model.
- Tokens in input ids are masked. We can mask a random selection of tokens now that we've produced a duplicate of input ids for labels.
- Make a loss calculation. We use our model to process the input ids and labels tensors and determine the loss between them.

25. What is the meaning of Pragmatic Analysis in NLP?

Pragmatic Analysis is concerned with outside word knowledge, which refers to information that is not contained in the documents and/or questions. The many parts of the language that require real-world knowledge are derived from a pragmatics analysis that focuses on what was described and reinterpreted by what it truly meant.

26. What is the meaning of N-gram in NLP?

Text N-grams are commonly used in text mining and natural language processing. They're essentially a collection of co-occurring words within a specific frame, and when computing the n-grams, you usually advance one word (although you can move X words forward in more advanced scenarios).

27. What do you mean by perplexity in NLP?

It's a statistic for evaluating the effectiveness of language models. It is described mathematically as a function of the likelihood that the language model describes a test sample. The perplexity of a test sample X = x1, x2, x3,....,xn is given by,

PP(X)=P(x1,x2,...,xN)-1N

The total number of word tokens is N.

The more perplexing the situation, the less information the language model conveys.

Conclusion

One of the most important reasons for NLP is that it allows computers to converse with people in natural language. Other language-related activities are also scaled. Computers can now hear, analyse, quantify, and identify which parts of speech are significant thanks to Natural Language Processing (NLP). NLP has a wide range of applications, including chatbots, sentiment analysis, and market intelligence. Since its introduction, NLP has grown in popularity. Today, devices like Amazon's Alexa are extensively used all over the world. And, for businesses, business intelligence and consumer monitoring are quickly gaining traction and will soon rule the industry.

References and Resources:

- Natural Language Processing with Python – Book by Edward Loper, Ewan Klein, and Steven Bird (Published by: O'Reilly Media, Inc.)
- Practical Natural Language Processing – By Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana (Published by: O'Reilly Media, Inc.)
- Natural Language Processing in Action: Understanding, Analyzing, and Generating Text with Python – Book by Cole Howard, Hannes Hapke, and Hobson Lane

NLP MCQ

1.Which of the following are some of the use cases of NLP?

- ○ Text Summarization
- ○ Topic modeling
- ○ Information extraction
- ◉ All of the above

2.Which of the following are some of the components on NLP?

- Pragmatic analysis
- Entity extraction
- Syntactic analysis
- (•) All of the above

3.What is NER in Natural Language Processing?

- (•) Named Entity Recognition
- Named Entity Response
- Name Explicit Response
- None of the above

4.Which of the following techniques can be used to find the distance between 2-word vectors in NLP?

- N-grams
- (•) Euclidean distance
- Lemmatization
- All of the above

5.Which of the following are the keyword Normalization techniques in NLP?

- Part of Speech
- Named entity recognition
- (•) Lemmatization
- None of the above

6.Which of the following are some of the pre-processing techniques in NLP?

- Removal of stop words
- Converting to lowercase
- Stemming and Lemmatization
- (•) All of the above

7.Which of the following are NLP tools?

- Natural language Toolkit
- SpaCy
- CogcompNLP
- (•) All of the above

8.Information extraction in NLP includes which of the following?

- Network Graph Module
- Document Classification & Language Modeling Module
- (•) Both (a) and (b)
- None of the above

9.Which of the following are differences between NLTK and Spacey?

- NLTK can support word vectors while Spacey cannot
- NLTK supports a wider range of languages compared to Spacey (Spacey supports only 7 languages)
- While Spacey has a collection of programs to choose from, NLTK contains only the best-suited algorithm for a problem in its toolkit
- While NLTK has an object-oriented library, Spacey has a string processing library

10.Which of the following are types of text summarization?

- Abstraction-based summarization
- Extraction-based summarization
- Both (a) and (b)
- None of the above.

GREATLEARNING

(https://www.mygreatlearning.com/blog/nlp-interview-questions/)

NLP Interview Questions for Freshers

Are you ready to kickstart your NLP career? Start your professional career with these Natural Language Processing interview questions for freshers. We will start with the basics and move towards more advanced questions. If you are an experienced professional, this section will help you brush up your NLP skills.

1. What is Naive Bayes algorithm, When we can use this algorithm in NLP?

Naive Bayes algorithm is a collection of classifiers which works on the principles of the Bayes' theorem. This series of NLP model forms a family of algorithms that can be used for a wide range of classification tasks including sentiment prediction, filtering of spam, classifying documents and more.

Naive Bayes algorithm converges faster and requires less training data. Compared to other discriminative models like logistic regression, Naive Bayes model it takes lesser time to train. This algorithm is perfect for use while working with multiple classes and text classification where the data is dynamic and changes frequently.

2. Explain Dependency Parsing in NLP?

Dependency Parsing, also known as Syntactic parsing in NLP is a process of assigning syntactic structure to a sentence and identifying its dependency parses. This process is crucial to understand the correlations between the "head" words in the syntactic structure.
The process of dependency parsing can be a little complex considering how any sentence can have more than one dependency parses. Multiple parse trees are known as ambiguities. Dependency parsing needs to resolve these ambiguities in order to effectively assign a syntactic structure to a sentence.

Dependency parsing can be used in the semantic analysis of a sentence apart from the syntactic structuring.

3. What is text Summarization?

Text summarization is the process of shortening a long piece of text with its meaning and effect intact. Text summarization intends to create a summary of any given piece of text and outlines the main points of the document. This technique has improved in recent times and is capable of summarizing volumes of text successfully.

Text summarization has proved to a blessing since machines can summarise large volumes of text in no time which would otherwise be really time-consuming. There are two types of text summarization:

- Extraction-based summarization
- Abstraction-based summarization

4. What is NLTK? How is it different from Spacy?

NLTK or Natural Language Toolkit is a series of libraries and programs that are used for symbolic and statistical natural language processing. This toolkit contains some of the most powerful libraries that can work on different ML techniques to break down and understand human language. NLTK is used for Lemmatization, Punctuation, Character count, Tokenization, and Stemming. The difference between NLTK and Spacey are as follows:

- While NLTK has a collection of programs to choose from, Spacey contains only the best-suited algorithm for a problem in its toolkit
- NLTK supports a wider range of languages compared to Spacey (Spacey supports only 7 languages)
- While Spacey has an object-oriented library, NLTK has a string processing library
- Spacey can support word vectors while NLTK cannot

5. What is information extraction?

Information extraction in the context of Natural Language Processing refers to the technique of extracting structured information automatically from unstructured sources to ascribe meaning to it. This can include extracting information regarding attributes of entities, relationship between different entities and more. The various models of information extraction includes:

- Tagger Module
- Relation Extraction Module
- Fact Extraction Module
- Entity Extraction Module
- Sentiment Analysis Module
- Network Graph Module
- Document Classification & Language Modeling Module

6. What is Bag of Words?

Bag of Words is a commonly used model that depends on word frequencies or occurrences to train a classifier. This model creates an occurrence matrix for documents or sentences irrespective of its grammatical structure or word order.

7. What is Pragmatic Ambiguity in NLP?

Pragmatic ambiguity refers to those words which have more than one meaning and their use in any sentence can depend entirely on the context. Pragmatic ambiguity can result in multiple interpretations of the same sentence. More often than not, we come across sentences which have words with multiple meanings, making the sentence open to interpretation. This multiple interpretation causes ambiguity and is known as Pragmatic ambiguity in NLP.

8. What is Masked Language Model?

Masked language models help learners to understand deep representations in downstream tasks by taking an output from the corrupt input. This model is often used to predict the words to be used in a sentence.

9. What is the difference between NLP and CI(Conversational Interface)?

The difference between NLP and CI is as follows:

Natural Language Processing (NLP)	Conversational Interface (CI)
NLP attempts to help machines understand and learn how language concepts work.	CI focuses only on providing users with an interface to interact with.
NLP uses AI technology to identify, understand, and interpret the requests of users through language.	CI uses voice, chat, videos, images, and more such conversational aid to create the user interface.

10. What are the best NLP Tools?

Some of the best NLP tools from open sources are:

- SpaCy
- TextBlob
- Textacy
- Natural language Toolkit (NLTK)
- Retext
- NLP.js
- Stanford NLP
- CogcompNLP

11. What is POS tagging?

Parts of speech tagging better known as POS tagging refer to the process of identifying specific words in a document and grouping them as part of speech, based on its context. POS tagging is also known as grammatical tagging since it involves understanding grammatical structures and identifying the respective component.

POS tagging is a complicated process since the same word can be different parts of speech depending on the context. The same general process used for word mapping is quite ineffective for POS tagging because of the same reason.

12. What is NES?

Name entity recognition is more commonly known as NER is the process of identifying specific entities in a text document that are more informative and have a unique context. These often denote places, people, organizations, and more. Even though it seems like these entities are proper nouns, the NER process is far from identifying just the nouns. In fact, NER involves entity chunking or extraction wherein entities are segmented to categorize them under different predefined classes. This step further helps in extracting information.

NLP Interview Questions for Experienced

13. Which of the following techniques can be used for keyword normalization in NLP, the process of converting a keyword into its base form?

a. Lemmatization
b. Soundex
c. Cosine Similarity
d. N-grams

Answer: a)

Lemmatization helps to get to the base form of a word, e.g. are playing -> play, eating -> eat, etc. Other options are meant for different purposes.

14. Which of the following techniques can be used to compute the distance between two-word vectors in NLP?

a. Lemmatization
b. Euclidean distance
c. Cosine Similarity
d. N-grams

Answer: b) and c)

Distance between two-word vectors can be computed using Cosine similarity and Euclidean Distance. Cosine Similarity establishes a cosine angle between the vector of two words. A cosine angle close to each other between two-word vectors indicates the words are similar and vice versa.

E.g. cosine angle between two words “Football” and “Cricket” will be closer to 1 as compared to the angle between the words “Football” and “New Delhi”.

Python code to implement CosineSimlarity function would look like this:

```
def cosine_similarity(x,y):

   return np.dot(x,y)/( np.sqrt(np.dot(x,x)) * np.sqrt(np.dot(y,y)) )

q1 = wikipedia.page('Strawberry')

q2 = wikipedia.page('Pineapple')

q3 = wikipedia.page('Google')

q4 = wikipedia.page('Microsoft')
```

```
cv = CountVectorizer()

X = np.array(cv.fit_transform([q1.content, q2.content, q3.content, q4.content]).todense())

print ("Strawberry Pineapple Cosine Distance", cosine_similarity(X[0],X[1]))

print ("Strawberry Google Cosine Distance", cosine_similarity(X[0],X[2]))

print ("Pineapple Google Cosine Distance", cosine_similarity(X[1],X[2]))

print ("Google Microsoft Cosine Distance", cosine_similarity(X[2],X[3]))

print ("Pineapple Microsoft Cosine Distance", cosine_similarity(X[1],X[3]))
```

```
Strawberry Pineapple Cosine Distance 0.8899200413701714

Strawberry Google Cosine Distance 0.7730935582847817

Pineapple Google Cosine Distance 0.789610214147025

Google Microsoft Cosine Distance 0.8110888282851575
```

Usually Document similarity is measured by how close semantically the content (or words) in the document are to each other. When they are close, the similarity index is close to 1, otherwise near 0.

The **Euclidean distance** between two points is the length of the shortest path connecting them. Usually computed using Pythagoras theorem for a triangle.

15. What are the possible features of a text corpus in NLP?

a. Count of the word in a document
b. Vector notation of the word
c. Part of Speech Tag
d. Basic Dependency Grammar
e. All of the above

Answer: e)

All of the above can be used as features of the text corpus.

16. You created a document term matrix on the input data of 20K documents for a Machine learning model. Which of the following can be used to reduce the dimensions of data?

1. Keyword Normalization
2. Latent Semantic Indexing

3. Latent Dirichlet Allocation

a. only 1
b. 2, 3
c. 1, 3
d. 1, 2, 3

Answer: d)

17. Which of the text parsing techniques can be used for noun phrase detection, verb phrase detection, subject detection, and object detection in NLP.

a. Part of speech tagging
b. Skip Gram and N-Gram extraction
c. Continuous Bag of Words
d. Dependency Parsing and Constituency Parsing

Answer: d)

18. Dissimilarity between words expressed using cosine similarity will have values significantly higher than 0.5

a. True
b. False

Answer: a)

19. Which one of the following is keyword Normalization techniques in NLP

a. Stemming
b. Part of Speech
c. Named entity recognition
d. Lemmatization

Answer: a) and d)

Part of Speech (POS) and Named Entity Recognition(NER) is not keyword Normalization techniques. Named Entity helps you extract Organization, Time, Date, City, etc., type of entities from the given sentence, whereas Part of Speech helps you extract Noun, Verb, Pronoun, adjective, etc., from the given sentence tokens.

20. Which of the below are NLP use cases?

a. Detecting objects from an image
b. Facial Recognition
c. Speech Biometric
d. Text Summarization

Ans: d)

a) And b) are Computer Vision use cases, and c) is the Speech use case.
Only d) Text Summarization is an NLP use case.

21. In a corpus of N documents, one randomly chosen document contains a total of T terms and the term “hello” appears K times.

What is the correct value for the product of TF (term frequency) and IDF (inverse-document-frequency), if the term “hello” appears in approximately one-third of the total documents?
a. KT * Log(3)
b. T * Log(3) / K
c. K * Log(3) / T
d. Log(3) / KT

Answer: (c)

formula for TF is K/T
formula for IDF is log(total docs / no of docs containing “data”)
= log(1 / (⅓))
= log (3)

Hence, the correct choice is Klog(3)/T

22. In NLP, The algorithm decreases the weight for commonly used words and increases the weight for words that are not used very much in a collection of documents

a. Term Frequency (TF)
b. Inverse Document Frequency (IDF)
c. Word2Vec
d. Latent Dirichlet Allocation (LDA)

Answer: b)

23. In NLP, The process of removing words like “and”, “is”, “a”, “an”, “the” from a sentence is called as

a. Stemming
b. Lemmatization
c. Stop word
d. All of the above

Ans: c)

In Lemmatization, all the stop words such as a, an, the, etc.. are removed. One can also define custom stop words for removal.

24. In NLP, The process of converting a sentence or paragraph into tokens is referred to as Stemming

a. True
b. False

Answer: b)

The statement describes the process of tokenization and not stemming, hence it is False.

25. In NLP, Tokens are converted into numbers before giving to any Neural Network

a. True
b. False

Answer: a)

In NLP, all words are converted into a number before feeding to a Neural Network.

26. Identify the odd one out

a. nltk
b. scikit learn
c. SpaCy
d. BERT

Answer: d)

All the ones mentioned are NLP libraries except BERT, which is a word embedding.

27. TF-IDF helps you to establish?

a. most frequently occurring word in document
b. the most important word in the document

Answer: b)

TF-IDF helps to establish how important a particular word is in the context of the document corpus. TF-IDF takes into account the number of times the word appears in the document and is offset by the number of documents that appear in the corpus.

- TF is the frequency of terms divided by the total number of terms in the document.
- IDF is obtained by dividing the total number of documents by the number of documents containing the term and then taking the logarithm of that quotient.
- Tf.idf is then the multiplication of two values TF and IDF.

Suppose that we have term count tables of a corpus consisting of only two documents, as listed here:

Term	Document 1 Frequency	Document 2 Frequency
This	1	1
is	1	1
a	2	
Sample	1	
another		2
example		3

The calculation of tf–idf for the term “this” is performed as follows:

for "this"

---

tf("this", d1) = 1/5 = 0.2

tf("this", d2) = 1/7 = 0.14

idf("this", D) = log (2/2) =0

hence tf-idf

tfidf("this", d1, D) = 0.2* 0 = 0

tfidf("this", d2, D) = 0.14* 0 = 0

for "example"

---

tf("example", d1) = 0/5 = 0

tf("example", d2) = 3/7 = 0.43

idf("example", D) = log(2/1) = 0.301

tfidf("example", d1, D) = tf("example", d1) * idf("example", D) = 0 * 0.301 = 0

tfidf("example", d2, D) = tf("example", d2) * idf("example", D) = 0.43 * 0.301 = 0.129

In its raw frequency form, TF is just the frequency of the "this" for each document. In each document, the word "this" appears once; but as document 2 has more words, its relative frequency is smaller.

An IDF is constant per corpus, and accounts for the ratio of documents that include the word "this". In this case, we have a corpus of two documents and all of them include the word "this". So TF–IDF is zero for the word "this", which implies that the word is not very informative as it appears in all documents.

The word "example" is more interesting – it occurs three times, but only in the second document. To understand more about NLP, check out these NLP projects.

28. In NLP, The process of identifying people, an organization from a given sentence, paragraph is called

a. Stemming
b. Lemmatization
c. Stop word removal
d. Named entity recognition

Answer: d)

29. Which one of the following is not a pre-processing technique in NLP

a. Stemming and Lemmatization
b. converting to lowercase
c. removing punctuations
d. removal of stop words
e. Sentiment analysis

Answer: e)

Sentiment Analysis is not a pre-processing technique. It is done after pre-processing and is an NLP use case. All other listed ones are used as part of statement pre-processing.

30. In text mining, converting text into tokens and then converting them into an integer or floating-point vectors can be done using

a. CountVectorizer
b. TF-IDF
c. Bag of Words
d. NERs

Answer: a)

CountVectorizer helps do the above, while others are not applicable.

```
text =["Rahul is an avid writer, he enjoys studying understanding and presenting. He loves to play"]

vectorizer = CountVectorizer()

vectorizer.fit(text)

vector = vectorizer.transform(text)

print(vector.toarray())
```

Output

```
[[1 1 1 1 2 1 1 1 1 1 1 1 1 1]]
```

The second section of the interview questions covers advanced NLP techniques such as Word2Vec, GloVe word embeddings, and advanced models such as GPT, Elmo, BERT, XLNET-based *questions, and explanations.*

31. In NLP, Words represented as vectors are called Neural Word Embeddings

a. True
b. False

Answer: a)

Word2Vec, GloVe based models build word embedding vectors that are multidimensional.

32. In NLP, Context modeling is supported with which one of the following word embeddings

1. a. Word2Vec
2. b) GloVe
3. c) BERT
4. d) All of the above

Answer: c)

Only BERT (Bidirectional Encoder Representations from Transformer) supports context modelling where the previous and next sentence context is taken into consideration. In Word2Vec, GloVe only word embeddings are considered and previous and next sentence context is not considered.

33. In NLP, Bidirectional context is supported by which of the following embedding

a. Word2Vec
b. BERT
c. GloVe
d. All the above

Answer: b)

Only BERT provides a bidirectional context. The BERT model uses the previous and the next sentence to arrive at the context.Word2Vec and GloVe are word embeddings, they do not provide any context.

34. Which one of the following Word embeddings can be custom trained for a specific subject in NLP

a. Word2Vec
b. BERT
c. GloVe
d. All the above

Answer: b)

BERT allows Transform Learning on the existing pre-trained models and hence can be custom trained for the given specific subject, unlike Word2Vec and GloVe where existing word embeddings can be used, no transfer learning on text is possible.

35. Word embeddings capture multiple dimensions of data and are represented as vectors

a. True
b. False

Answer: a)

36. In NLP, Word embedding vectors help establish distance between two tokens

a. True
b. False

Answer: a)

One can use Cosine similarity to establish the distance between two vectors represented through Word Embeddings

37. Language Biases are introduced due to historical data used during training of word embeddings, which one amongst the below is not an example of bias

a. New Delhi is to India, Beijing is to China
b. Man is to Computer, Woman is to Homemaker

Answer: a)

Statement b) is a bias as it buckets Woman into Homemaker, whereas statement a) is not a biased statement.

38. Which of the following will be a better choice to address NLP use cases such as semantic similarity, reading comprehension, and common sense reasoning

a. ELMo
b. Open AI's GPT
c. ULMFit

Answer: b)

Open AI's GPT is able to learn complex patterns in data by using the Transformer models Attention mechanism and hence is more suited for complex use cases such as semantic similarity, reading comprehensions, and common sense reasoning.

39. Transformer architecture was first introduced with?

a. GloVe
b. BERT
c. Open AI's GPT
d. ULMFit

Answer: c)

ULMFit has an LSTM based Language modeling architecture. This got replaced into Transformer architecture with Open AI's GPT.

40. Which of the following architecture can be trained faster and needs less amount of training data

a. LSTM-based Language Modelling
b. Transformer architecture

Answer: b)

Transformer architectures were supported from GPT onwards and were faster to train and needed less amount of data for training too.

41. Same word can have multiple word embeddings possible with _____?

a. GloVe
b. Word2Vec
c. ELMo
d. nltk

Answer: c)

EMLo word embeddings support the same word with multiple embeddings, this helps in using the same word in a different context and thus captures the context than just the meaning of the word unlike in GloVe and Word2Vec. Nltk is not a word embedding.

42. For a given token, its input representation is the sum of embedding from the token, segment and position

embedding
a. ELMo
b. GPT
c. BERT
d. ULMFit
Answer: c)
BERT uses token, segment and position embedding.

43. Trains two independent LSTM language model left to right and right to left and shallowly concatenates them.

a. GPT
b. BERT
c. ULMFit
d. ELMo
Answer: d)
ELMo tries to train two independent LSTM language models (left to right and right to left) and concatenates the results to produce word embedding.

44. Uses unidirectional language model for producing word embedding.

a. BERT
b. GPT
c. ELMo
d. Word2Vec

Answer: b)

GPT is a bidirectional model and word embedding is produced by training on information flow from left to right. ELMo is bidirectional but shallow. Word2Vec provides simple word embedding.

45. In this architecture, the relationship between all words in a sentence is modelled irrespective of their position. Which architecture is this?

a. OpenAI GPT
b. ELMo
c. BERT
d. ULMFit

Ans: c)

BERT Transformer architecture models the relationship between each word and all other words in the sentence to generate attention scores. These attention scores are later used as weights for a weighted average of all words' representations which is fed into a fully-connected network to generate a new representation.

46. List 10 use cases to be solved using NLP techniques?

- Sentiment Analysis
- Language Translation (English to German, Chinese to English, etc..)
- Document Summarization
- Question Answering
- Sentence Completion
- Attribute extraction (Key information extraction from the documents)
- Chatbot interactions
- Topic classification
- Intent extraction
- Grammar or Sentence correction

- Image captioning
- Document Ranking
- Natural Language inference

47. Transformer model pays attention to the most important word in Sentence.

a. True
b. False

Ans: a) Attention mechanisms in the Transformer model are used to model the relationship between all words and also provide weights to the most important word.

48. Which NLP model gives the best accuracy amongst the following?

a. BERT
b. XLNET
c. GPT-2
d. ELMo

Ans: b) XLNET

XLNET has given best accuracy amongst all the models. It has outperformed BERT on 20 tasks and achieves state of art results on 18 tasks including sentiment analysis, question answering, natural language inference, etc.

49. Permutation Language models is a feature of

a. BERT
b. EMMo
c. GPT
d. XLNET

Ans: d)

XLNET provides permutation-based language modelling and is a key difference from BERT. In permutation language modeling, tokens are predicted in a random manner and not sequential. The order of prediction is not necessarily left to right and can be right to left. The original order of words is not changed but a prediction can be random. The conceptual difference between BERT and XLNET can be seen from the following diagram.

50. Transformer XL uses relative positional embedding

a. True
b. False

Ans: a)

Instead of embedding having to represent the absolute position of a word, Transformer XL uses an embedding to encode the relative distance between the words. This embedding is used to compute the attention score between any 2 words that could be separated by n words before or after.

There, you have it – all the probable questions for your NLP interview. Now go, give it your best shot.

Natural Language Processing FAQs

1. Why do we need NLP?

One of the main reasons why NLP is necessary is because it helps computers communicate with humans in natural language. It also scales other language-related tasks. Because of NLP, it is possible for computers to hear speech, interpret this speech, measure it and also determine which parts of the speech are important.

2. What must a natural language program decide?

A natural language program must decide what to say and when to say something.

3. Where can NLP be useful?

NLP can be useful in communicating with humans in their own language. It helps improve the efficiency of the machine translation and is useful in emotional analysis too. It can be helpful in sentiment analysis using python too. It also helps in structuring highly unstructured data. It can be helpful in creating chatbots, Text Summarization and virtual assistants.

4. How to prepare for an NLP Interview?

The best way to prepare for an NLP Interview is to be clear about the basic concepts. Go through blogs that will help you cover all the key aspects and remember the important topics. Learn specifically for the interviews and be confident while answering all the questions.

5. What are the main challenges of NLP?

Breaking sentences into tokens, Parts of speech tagging, Understanding the context, Linking components of a created vocabulary, and Extracting semantic meaning are currently some of the main challenges of NLP.

6. Which NLP model gives best accuracy?

Naive Bayes Algorithm has the **highest accuracy** when it comes to NLP models. It gives up to 73% correct predictions.

7. What are the major tasks of NLP?

Translation, named entity recognition, relationship extraction, sentiment analysis, speech recognition, and topic segmentation are few of the major tasks of NLP. Under unstructured data, there can be a lot of untapped information that can help an organization grow.

8. What are stop words in NLP?

Common words that occur in sentences that add weight to the sentence are known as stop words. These stop words act as a bridge and ensure that sentences are grammatically correct. In simple terms, words that are

filtered out before processing natural language data is known as a stop word and it is a common pre-processing method.

9. What is stemming in NLP?

The process of obtaining the root word from the given word is known as stemming. All tokens can be cut down to obtain the root word or the stem with the help of efficient and well-generalized rules. It is a rule-based process and is well-known for its simplicity.

10. Why is NLP so hard?

There are several factors that make the process of Natural Language Processing difficult. There are hundreds of natural languages all over the world, words can be ambiguous in their meaning, each natural language has a different script and syntax, the meaning of words can change depending on the context, and so the process of NLP can be difficult. If you choose to upskill and continue learning, the process will become easier over time.

11. What does a NLP pipeline consist of *?

The overall architecture of an **NLP pipeline consists** of several layers: a user interface; one or several **NLP** models, depending on the use case; a Natural Language Understanding layer to describe the **meaning of** words and sentences; a preprocessing layer; microservices for linking the components together and of course.

12. How many steps of NLP is there?

The five phases of NLP involve lexical (structure) analysis, parsing, semantic analysis, discourse integration, and pragmatic analysis.

Further Reading

1. Python Interview Questions and Answers for 2022
2. Machine Learning Interview Questions and Answers for 2022
3. 100 Most Common Business Analyst Interview Questions
4. Artificial Intelligence Interview Questions for 2022 | AI Interview Questions
5. 100+ Data Science Interview Questions for 2022
6. Common Interview Questions