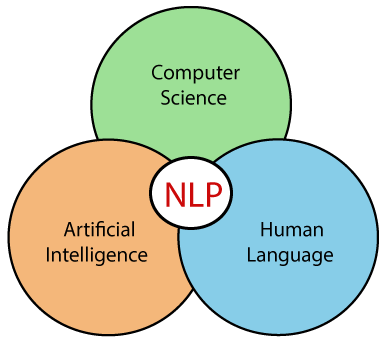
07-06-2023

**INTRODUCTION TO NATURAL LANGUAGE PROCESSING**

Natural language processing (NLP) refers to the branch of computer science, specifically the branch of artificial intelligence.

This is concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. NLP combines computational linguistics—rule-based modeling of human language—with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to ‘understand’ its full meaning, complete with the speaker or writer’s intent and sentiment.

* The field of study that focuses on the interactions between human language and computers is called Natural Language Processing.
* It enables computers to understand and process human language.
* The main intention of NLP is to build systems that are able to make sense of text and execute certain tasks like spell-check, text translation, topic classification, etc.



**Components of NLP:**

There are two components of NLP

* **Natural language generation** (NLG) and **Natural language understanding** (NLU).
* **Natural Language Understanding (NLU)**

It will help the machine to understand and analyze human language by extracting the metadata.

It is used to map the given input into a useful representation.

It is used to analyze different aspects of the language.

**Natural language understanding:**

NLP is difficult to implement - data contains ambiguity and uncertainty.

**Ambiguity:**

Ability of being understood in more than one form. (Capability of being understood more than one form)

Natural language is very ambiguous. NLP has the following types of ambiguities.

1. **Lexical ambiguity**: The ambiguity of a single word is called lexical ambiguity. For example, treating the word silver as a noun, an adjective, or a verb.
2. **Syntactic Ambiguity**: This kind of ambiguity occurs when a sentence is parsed in different ways. For example, the sentence “Raghav saw the girl with the telescope”. It is ambiguous whether the Raghav saw the girl carrying a telescope or he saw her through his telescope.
3. **Referential ambiguity**: Referential Ambiguity exists when you are referring to something using the pronoun.

**Natural Language Generation:**

It will act like a translator that convert the computerized data into natural language representation. It mainly involves Text planning, Sentence planning, and Text Realization.

* Text planning: It includes retrieving the relevant content from knowledge base.
* Sentence planning: It includes choosing required words, forming meaningful phrases, setting tone of the sentence.
* Text Realization: It is mapping sentence plan into sentence structure.

**Pipeline of NLP:**

1. Sentence Segmentation
2. Word tokenization
3. Stemming
4. Lemmatization
5. Stop word analysis
6. Dependency parsing
7. Parts of speech tagging
8. **Sentence Segmentation:** Sentence Segment is the first step for building the NLP pipeline. It breaks the paragraph into separate sentences.

Example:

Independence Day is one of the important festivals for every Indian citizen. It is celebrated on the 15th of August each year ever since India got independence from the British rule. The day celebrates independence in the true sense.

*Sentence Segment produces the following result:*

"Independence Day is one of the important festivals for every Indian citizen."

"It is celebrated on the 15th of August each year ever since India got independence from the British rule."

"This day celebrates independence in the true sense."

1. **Word Tokenization**: It is used to break the sentence into separate words or tokens.

Example:

Hi how are you? Nice to meet you

[“hi”,” how”, “are” ……………………” meet”, “you”]

1. **Stemming**: Stemming is used to normalize words into its base form or root form.

Example:

celebrates, celebrated and celebrating, all these words are originated with a single root word "celebrate."

1. **Lemmatization:** Like stemming but it will produce the root word which has a real English meaning.

Example:

In lemmatization, the words intelligence, intelligent, and intelligently has a root word intelligent, which has a meaning.

1. **Stop words analysis:** Stop words – very high frequent words in English – **“is”, “the”** , and **“a”.** The process of filtering out the stop words.

Example:

He is a brave boy.

1. **Dependency Parsing:** Dependency Parsing is used to find that how all the words in the sentence are related to each other.

POS-Tags:

POS – Part of speech, which will include Noun, verb, adverb, adjective.

* Noun-n
* Verb-v
* Adjective-a
* Adverb-r

**Phases of NLP:**

Lexical Analysis------🡪 Syntactic Analysis---🡪 Semantic Analysis----🡪 Discourse Integration----🡪 Pragmatic Analysis

1. Lexical Analysis:

* The first phase of NLP is the Lexical Analysis.
* It involves identifying and analysing the structure of chunks.
* Lexical analysis is dividing the whole chunk of txt into paragraphs, sentences, and words.
* Stemming and lemmatization takes place in this phase.

1. Syntactic Analysis:

* This is also called as Parsing.
* Syntactic Analysis is used to check grammar, word arrangements, and shows the relationship among the words.
* Example: Chennai goes to the EDA.
* This sentence above does not make any sense, so this sentence is rejected by the Syntactic analyser.

1. Semantic Analysis:

* Determines the possible meanings of a sentence.
* Semantic analysis is concerned with the meaning representation.
* It mainly focusses on meaning of words, phrases, and sentences.

**TF-IDF VECTORIZER:**

Now, the problem with natural language processing is that the data is in the form of raw text, so that the text need to be transformed into the vector.

* The process of transforming text into a vector is called as text vectorization.
* Text vectorization algorithm namely TF-IDF vectorizer, which is a very popular approach for traditional machine learning algorithms can help in transforming text into vectors.
* Term frequency-inverse document frequency is a text vectorizer that transforms the text into a usable vector. It combines 2 concepts, Term Frequency (TF) and Document Frequency (DF).

1. Term frequency:

The term frequency is the number of occurrences of a specific term in a document.

Term frequency represents every text from the data as a matrix whose rows are the number of documents

and columns are the number of distinct terms throughout all documents.

|  |  |
| --- | --- |
| Text1 | I love natural language processing but I hate python |
| Text2 | I like image processing |
| Text3 | I like signal processing and image processing |

Solution:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | and | but | hate | I | image | language | like | love | natural | processing | python | signal |
| Text 1 | 0 | 1 | 1 | 2 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| Text 2 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| Text 3 | 1 | 0 |  | 1 | 1 | 0 | 1 | 0 | 0 | 2 | 0 | 1 |

1. Inverse Document Frequency: (IDF)

It is the weight of the term.

* + - idfᵢ - IDF score for term I
    - dfᵢ - The number of documents containing term I
    - n - The total number of documents.
    - Higher the df of the term lower the IDF
    - When the number of DF is equal to n which means that the term appears in all documents, the IDF will
    - be zero, since log (1) is zero.

**TF-IDF:**

The TF-IDF score as the name suggests is just a multiplication of the term frequency matrix with its IDF, it can be calculated as follow: wᵢⱼ is TF-IDF score for term i in document j, tfᵢⱼ is term frequency for term i in document j, and idfᵢ is IDF score for term i.

***W (i, j) = tf (i, j) X idf(i)***