

Natural Language Processing (NLP) Tutorial

Learn the basics and advanced concepts of natural language processing (NLP) with our complete NLP tutorial and get ready to explore the vast and exciting field of NLP, where technology meets human language.

NLP tutorial is designed for both beginners and professionals. Whether you're a data scientist, a developer, or someone curious about the power of language, our tutorial will provide you with the knowledge and skills you need to take your understanding of NLP to the next level.

What is NLP?

NLP stands for Natural Language Processing. It is the branch of Artificial Intelligence that gives the ability to machine understand and process human languages. Human languages can be in the form of text or audio format.

History of NLP

Natural Language Processing started in 1950 When **Alan Mathison Turing** published an article in the name **Computing Machinery and Intelligence**. It is based on Artificial intelligence. It talks about automatic interpretation and generation of natural language. As the technology evolved, different approaches have come to deal with NLP tasks.

- **Heuristics-Based NLP:** This is the initial approach of NLP. It is based on defined rules. Which comes from domain knowledge and expertise. Example: regex
- **Statistical Machine learning-based NLP:** It is based on statistical rules and machine learning algorithms. In this approach, algorithms are applied to the data and learned from the data, and applied to various tasks. Examples: Naive Bayes, support vector machine (SVM), hidden Markov model (HMM), etc.
- **Neural Network-based NLP:** This is the latest approach that comes with the evaluation of neural network-based learning, known as Deep learning. It provides good accuracy, but it is a very data-hungry and time-consuming approach. It requires high computational power to train the model. Furthermore, it is based on neural network architecture. Examples: Recurrent neural networks (RNNs), Long short-term memory networks (LSTMs), Convolutional neural networks (CNNs), Transformers, etc.

Components of NLP

There are two components of Natural Language Processing:

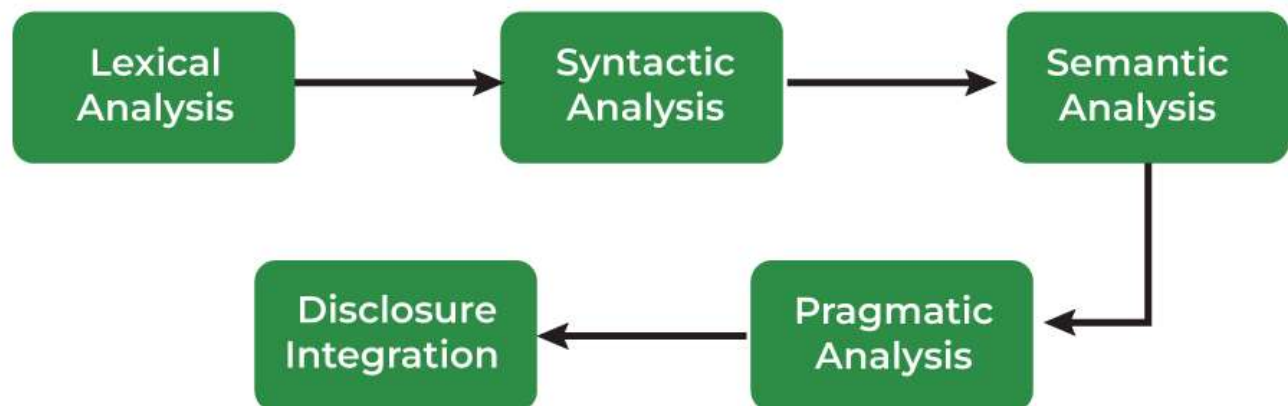
- Natural Language Understanding
- Natural Language Generation

Applications of NLP

The applications of Natural Language Processing are as follows:

- Text and speech processing like-Voice assistants – Alexa, Siri, etc.
- Text classification like Grammarly, Microsoft Word, and Google Docs
- Information extraction like-Search engines like DuckDuckGo, Google
- Chatbot and Question Answering like:- website bots
- Language Translation like:- Google Translate
- Text summarization

Phases of Natural Language Processing



NLP Libraries

- [NLTK](#)
- [Spacy](#)
- [Gensim](#)
- [fastText](#)
- [Stanford toolkit \(Glove\)](#)
- [Apache OpenNLP](#)

Classical Approaches

Classical Approaches to Natural Language Processing

- Text Preprocessing
 - Regular Expressions
 - [How to write Regular Expressions?](#)
 - [Properties of Regular expressions](#)
 - Text Preprocessing using RE
 - [Regular Expression](#)
 - [Email Extraction using RE](#)
 - Tokenization
 - [White Space Tokenization](#)
 - [Dictionary Based Tokenization](#)
 - [Rule-Based Tokenization](#)
 - [Regular Expression Tokenizer](#)
 - [Penn Treebank Tokenization](#)
 - [Spacy Tokenizer](#)
 - [Subword Tokenization](#)
 - [Tokenization with Textblob](#)
 - [Tokenize text using NLTK in python](#)
 - [How tokenizing text, sentences, and words works](#)
 - [Lemmatization](#)
 - [Stemming](#)
 - Types
 - [Porter Stemmer](#)
 - [Lovins Stemmer](#)
 - [Dawson Stemmer](#)
 - [Krovetz Stemmer](#)
 - [Xerox Stemmer](#)
 - Stopwords removal
 - [Removing stop words with NLTK in Python](#)
 - Parts of Speech (POS)
 - [Part of Speech – Default Tagging](#)
 - [Part of speech tagging – word corpus](#)
 - [Part of Speech Tagging with Stop words using NLTK in python](#)

- [Part of Speech Tagging using TextBlob](#)
- [Text Normalization](#)
- **Text Vectorization or Encoding:**
 - [vector space model \(VSM\)](#)
 - Words and vectors
 - [Cosine similarity](#)
 - Basic Text Vectorization approach:
 - [One-Hot Encoding](#)
 - [Byte-Pair Encoding \(BPE\)](#)
 - [Bag of words \(BOW\)](#)
 - [N-Grams](#)
 - [Term frequency Inverse Document Frequency \(TFIDF\)](#)
 - [N-Gram Language Modelling with NLTK](#)
 - Distributed Representations:
 - Word Embeddings
 - Pre-Trained Word Embeddings
 - [Word Embedding using Word2Vec](#)
 - [Finding the Word Analogy from given words using Word2Vec embeddings](#)
 - [GloVe](#)
 - [fasttext](#)
 - Train Own Word Embeddings
 - [Continuous bag of words \(CBOW\)](#)
 - [SkipGram](#)
 - [Doc2Vec](#)
 - Universal Text Representations
 - [Embeddings from Language Models \(ELMo\)](#)
 - [Bidirectional Encoder Representations from Transformers \(BERT\)](#)
 - Embeddings Visualizations
 - [t-sne \(t-distributed Stochastic Neighbouring Embedding\)](#)
 - [TextEvaluator](#)
 - Embeddings semantic properties
- **Semantic Analysis**
 - [What is Sentiment Analysis?](#)
 - [Understanding Semantic Analysis](#)
 - Sentiment classification:
 - [Naive Bayes Classifiers](#)

- Logistic Regression
- [Sentiment Classification Using BERT](#)
- [Twitter Sentiment Analysis using textblob](#)
- **Parts of Speech tagging and Named Entity Recognizations:**
 - [Parts of Speech tagging with NLTK](#)
 - [Parts of Speech tagging with spacy](#)
 - Hidden Markov Model for POS tagging
 - [Markov Chains](#)
 - [Hidden Markov Model](#)
 - [Viterbi Algorithm](#)
 - Conditional Random Fields (CRFs)
 - Conditional Random Fields (CRFs) for POS tagging
 - Named Entity Recognition
 - Rule Based Approach
 - [Named Entity Recognizations](#)
- **Neural Network for NLP:**
 - Feedforwards networks for NLP
 - [Recurrent Neural Networks](#)
 - RNN for Text Classifications
 - RNN for Sequence Labeling
 - Stacked RNNs
 - Bidirectional RNNs
 - [Long Short-Term Memory \(LSTM\)](#)
 - [LSTM with Tensorflow](#)
 - Bidirectional LSTM
 - [Gated Recurrent Unit \(GRU\)](#)
 - [Sentiment Analysis with RNN,LSTM, GRU](#)
 - [Emotion Detection using Bidirectional LSTM & GRU](#)
 - [Transformers for NLP](#)
- **Transfer Learning for NLP:**
 - [Bidirectional Encoder Representations from Transformers](#)
 - [RoBERTa](#)
 - [SpanBERT](#)
 - Transfer Learning with Fine-tuning
- **Informations Extractions**
 - Keyphrase Extraction

- Named Entity Recognition
- Relationship Extraction
- Information Retrieval
- Text Generations
 - [Text Generations introductions](#)
- Text summarization
 - Extractive Text Summarization using Gensim
- Questions – Answering
- Chatbot & Dialogue Systems:
 - [Simple Chat Bot using ChatterBot](#)
 - [GUI chat application using Tkinter](#)
- Machine translation
 - [Machine translation Introductions](#)
 - [Statistical Machine Translation Introduction](#)
- Phonetics
 - [Implement Phonetic Search in Python with Soundex Algorithm](#)
 - [Convert English text into the Phonetics](#)
- Speech Recognition and Text-to-Speech
 - [Convert Text to Speech](#)
 - [Convert Speech to text and text to Speech](#)
 - [Speech Recognition using Google Speech API](#)

Empirical and Statistical Approaches

- Treebank Annotation
- Fundamental Statistical Techniques for NLP
- Part-of-Speech Tagging
- Rules-based system
- Statistical Parsing
- Multiword Expressions
- Normalized Web Distance and Word Similarity
- Word Sense Disambiguation

FAQs on Natural Language Processing

What is the most difficult part of natural language processing?

Ambiguity is the main challenge of natural language processing because in natural language, words are unique, but they have different meanings depending upon the context which causes ambiguity on lexical, syntactic, and semantic levels.

What are the 4 pillars of NLP?

The four main pillars of NLP are 1.) Outcomes, 2.) Sensory acuity, 3.) behavioural flexibility, and 4.) report.

What language is best for natural language processing?

Python is considered the best programming language for NLP because of their numerous libraries, simple syntax, and ability to easily integrate with other programming languages.

What is the life cycle of NLP?

There are four stages included in the life cycle of NLP – development, validation, deployment, and monitoring of the models.