What is NLP Natural Language Processing?

**Natural Language Processing (NLP)** is a sub field of AI that enables computers to understand, interpret and generate human language. It basically is a combination of linguistics along with Machine Learning and Deep Learning to process human speech and enabling communication between machines and humans.

**Key tasks in NLP:**

* Text preprocessing (tokenization, stemming, lemmatization)
* Named Entity Recognition (NER)
* Part-of-Speech (POS) tagging
* Sentiment analysis
* Machine translation
* Text summarization
* Speech recognition

NLP is widely used in chatbots, search engines, and automated translation systems.

**Key Challenges in NLP?**

Training an NLP model comes with certain challenges to interpret the natural language we use.

1. **Ambiguity** – Words and sentences can have multiple meanings.
2. **Context Understanding** – Difficulty in capturing sarcasm, idioms, and implied meanings.
3. **Data Sparsity** – Limited labeled data for low-resource languages or domain-specific applications.
4. **Long-Range Dependencies** – Difficulty in handling context over long sentences or paragraphs.
5. **Out-of-Vocabulary (OOV) Words** – Handling rare or new words not present in the training data.
6. **Computational Complexity** – High resource requirements for training deep learning models.
7. **Bias and Fairness** – NLP models can inherit biases from training data, leading to unfair predictions.
8. **Ethical and Privacy Concerns** – Handling sensitive data responsibly to ensure user privacy and compliance.

**What are the main applications of NLP?**

machine translation, sentiment analysis, text summarization, speech recognition, chatbots and virtual assistants, information extraction, named entity recognition, text classification, and grammar checking.

**Explain the difference between structured and unstructured data in the context of NLP.**

In **NLP**, data can be categorized as **structured** or **unstructured** based on its organization and format:

1. **Structured Data:**
   * Well-organized and stored in a predefined format (e.g., tables, databases).
   * Contains clearly labeled features, making it easy to analyze.
   * Example in NLP: Sentiment-labeled datasets, chatbot logs with metadata (user ID, timestamp, message).
2. **Unstructured Data:**
   * Lacks a predefined format; free-form text without strict organization.
   * Requires preprocessing (tokenization, parsing) to extract meaningful information.
   * Example in NLP: Raw text from social media, emails, news articles, speech transcripts.

**What is tokenization in NLP? Why is it important?**

**Tokenization** is the process of splitting text into smaller units called **tokens** (words, subwords, or sentences). It basically is very crucial step in NLP.

**Importance of Tokenization:**

* **Prepares text for NLP models** by converting raw text into manageable units.
* **Enables feature extraction** by identifying meaningful linguistic elements.
* **Improves computational efficiency** by reducing text complexity.

**What are stop words, and why do we remove them during text preprocessing?**

**Stop words** are commonly used words (e.g., *the, is, and, of, in*) that carry little meaning and are often removed during text preprocessing to improve NLP model efficiency.

**Why Remove Stop Words?**

1. **Reduces Dimensionality** – Eliminates unnecessary words, making data processing faster.
2. **Improves Model Performance** – Focuses on meaningful words that contribute to context and meaning.
3. **Enhances Computational Efficiency** – Reduces the size of input data for NLP models.
4. **Removes Noise** – Prevents frequent, non-informative words from affecting analysis (e.g., in search engines and text classification).

However, **stop word removal depends on the task**—in some applications like sentiment analysis, certain stop words (e.g., *not, never*) may be important and should be retained.

**What is stemming and lemmatization? How do they differ?**

Stemming and lemmatization are both techniques used in Natural Language Processing (NLP) to reduce words to their base form, but the key difference is that stemming simply removes suffixes from words using heuristic rules(eg. Achieving 🡪 achiev), while lemmatization uses linguistic knowledge to find the correct base form of a word (called a lemma), considering its context and part of speech, ensuring the resulting word is a valid dictionary word;(eg. Achieving 🡪 achieve) making lemmatization more accurate but computationally heavier than stemming

The practical distinction between stemming and lemmatization is that, where stemming merely removes common suffixes from the end of word tokens, lemmatization ensures the output word is an existing normalized form of the word (for example, lemma) that can be found in the dictionary

**What is part-of-speech tagging in NLP?**

POS tagging is the process of labeling words in a sentence with their grammatical categories, such as nouns, verbs, adjectives, and adverbs.

POS tagging helps in understanding the context of words by identifying their grammatical roles. It resolves ambiguity and improves language comprehension.

* Helps in **syntactic and semantic analysis**.
* Improves **named entity recognition (NER)** and **sentiment analysis**.
* Assists in **word sense disambiguation** (e.g., *"book" as a noun vs. verb*).

POS tagging is typically performed using **rule-based, statistical, or deep learning models** (e.g., Hidden Markov Model, CRF, and Transformer-based models).

**What is Named Entity Recognition (NER)?**

Named Entity Recognition (NER) is an NLP task that involves identifying and classifying **named entities** in text into predefined categories such as people, organizations, locations, dates, and more.

**Example:**

* Text: *"Apple Inc. was founded by Steve Jobs in Cupertino on April 1st, 1976."*
* NER output:
  + *Apple Inc.* → Organization
  + *Steve Jobs* → Person
  + *Cupertino* → Location
  + *April 1st, 1976* → Date

**Importance of NER:**

* **Information Extraction:** Helps in extracting key details like company names, locations, and dates.
* **Data Structuring:** Converts unstructured text into structured data for further analysis.
* **Use Cases:** NER is used in search engines, automated summarization, and knowledge graph creation.

NER typically uses **machine learning** or **deep learning** models (e.g., CRF, LSTMs, or Transformers).

**What are n-grams in NLP?**

**What is the importance of word embeddings in NLP?**

What is the difference between CountVectorizer and TF-IDF?

What is Word2Vec, and how does it work?

What are some limitations of using bag-of-words models?

The **bag-of-words** (BoW) model is a simple and widely used technique for text representation, but it has several limitations:

1. **Lack of Context:**
   * BoW treats each word independently, ignoring **word order** and **context**. Words with similar meanings or those that should be related in context (e.g., *"bank"* as a financial institution vs. *"bank"* of a river) are treated the same.
2. **High Dimensionality:**
   * BoW generates a very large feature space, especially when the corpus contains a large vocabulary. This can result in **sparse vectors** and **computational inefficiency** (also known as the "curse of dimensionality").
3. **No Semantic Information:**
   * It fails to capture the **semantic meaning** of words. Words with similar meanings (e.g., *"happy"* and *"joyful"*) are represented as distinct features.
4. **Sensitivity to Rare Words:**
   * Rare words or words that appear once in a document might have high variance in the feature space but often do not contribute much to the document's overall meaning. This can result in **overfitting** or **noisy models**.
5. **Inability to Handle Synonyms and Polysemy:**
   * BoW cannot handle **synonymy** (e.g., *"car"* and *"vehicle"*) or **polysemy** (e.g., *"bat"* as a flying mammal vs. *"bat"* used in baseball). Each word is treated as an independent token.
6. **Over-representation of Frequent Words:**
   * Common words like *"the," "is," "in"* dominate the model, often overshadowing more meaningful words unless proper **stopword removal** is performed.
7. **Memory Intensive:**
   * BoW often requires storing large sparse matrices, which can be memory-intensive, especially for larger datasets.

In summary, while BoW is useful for simple tasks, it has **context and semantic limitations**, making it less effective for capturing complex relationships in text. More advanced models like **TF-IDF** and **word embeddings** address some of these issues.

What is Latent Dirichlet Allocation (LDA) and its use in topic modeling?

Topic modeling is a way of abstract modeling to discover the abstract ‘topics’ that occur in the collections of documents. The idea is that we will perform unsupervised classification on different documents, which find some natural groups in topics. We can answer the following question using topic modeling.

* What is the topic/main idea of the document?
* Given a document, can we find another document with a similar topic?
* How do topics field change over time?

Latent Dirichlet allocation is one of the most popular methods for performing topic modeling. Each document consists of various words and each topic can be associated with some words. The aim behind the LDA to find topics that the document belongs to, on the basis of words contains in it. It assumes that documents with similar topics will use a similar group of words.

What is syntactic parsing in NLP?

**Syntactic parsing** in NLP is the process of analyzing a sentence to understand its grammatical structure, identifying the syntactic relationships between words, and constructing a **parse tree** or **syntax tree**.

What is semantic analysis, and how does it differ from syntactic analysis?

**Semantic Analysis** in NLP focuses on **understanding the meaning** of words, phrases, and sentences in context. It involves interpreting the relationships between words in terms of their **semantic roles** and extracting the underlying meaning. Semantic analysis deals with resolving ambiguities related to word meanings and context.

**Word Sense Disambiguation (WSD):** Identifying the correct meaning of a word based on context (e.g., *bank* as a financial institution vs. *bank* of a river).

**Named Entity Recognition (NER):** Identifying and classifying entities (e.g., people, locations, dates) within text.

**Sentiment Analysis:** Determining the sentiment (positive, negative, neutral) expressed in text.

**Semantic Role Labeling (SRL):** Identifying the roles of words in a sentence (e.g., who did what to whom, when, and where).

**Syntactic analysis** is concerned with **how words are arranged** in a sentence, using grammatical rules (e.g., subject-verb agreement, word order).

**Semantic analysis** is concerned with **what the sentence means**—the underlying relationships and interpretation of the sentence based on word meanings and context.

How do you handle out-of-vocabulary (OOV) words in NLP?

Out-of-vocabulary (OOV) words refer to words that were not present in the model's training vocabulary, making them challenging for NLP models to process.

**To evaluate** an NLP model, you primarily use metrics like accuracy, precision, recall, F1-score, and depending on the task, specific metrics like BLEU (for machine translation) or ROUGE (for text summarization), comparing the model's output to a gold standard (human-annotated data) to assess how well it performs on tasks like text classification, sentiment analysis, or question answering;

* **Accuracy:** Simple metric, percentage of correct predictions, but can be misleading with imbalanced datasets.
* **Precision:** Proportion of positive predictions that are actually correct.
* **Recall:** Proportion of relevant cases that are correctly identified.
* **F1-score:** Harmonic mean of precision and recall, balances both aspects.
* **BLEU (Bilingual Evaluation Understudy):** Commonly used for machine translation, measures the overlap between generated and reference translations.
* **ROUGE (Recall-Oriented Understudy for Gisting Evaluation):** Often used for text summarization, measures how well the generated summary captures key information from the source text.

What is a language model?

A **language model** in NLP is a statistical model that is used to **predict the likelihood of a sequence of words** or characters in a sentence. It assigns a probability to a sequence of words based on the **context** provided by preceding words, helping machines understand and generate human language.

How do you preprocess text data for NLP tasks?

What is the Transformer model, and how has it revolutionized NLP?