NLP

NLP

Natural Language Processing (NLP) is a field of Artificial Intelligence (AI) that focuses on enabling machines to **understand, interpret, generate, and interact with human language.**

It's the bridge between **computers** and **human communication.**

🔷 Why NLP is challenging

Human language is:

- **Ambiguous** (same word can mean different things → "bank" = river bank / financial bank).
- **Contextual** (meaning depends on surrounding words).
- **Complex** (grammar, slang, sarcasm, cultural references).

🔷 Core Tasks in NLP

1. **Text Understanding**
   - Tokenization (splitting text into words/subwords).
   - Part-of-Speech tagging (noun, verb, adjective).
   - Named Entity Recognition (NER: names, places, dates).
   - Sentiment Analysis (positive/negative review).
2. **Text Representation**
   - Bag of Words, TF-IDF (traditional).
   - Word Embeddings (Word2Vec, GloVe).
   - Contextual embeddings (BERT, GPT).

3. **Text Generation**
    - Machine Translation (English → Hindi).
    - Chatbots.
    - Text summarization.
    - Question answering.

🔷 NLP Pipeline (Simplified)

Input text → Preprocessing → Feature Representation → Model (ML/DL) → Output (prediction, classification, generation).

Example:

- Input: "I love this movie!"
- Preprocessing: tokenization → [I, love, this, movie]
- Representation: embeddings (vectors)
- Model: sentiment classifier
- Output: "Positive"

🔷 Popular Deep Learning Architectures in NLP

- **RNN / LSTM / GRU** → Early sequence models.
- **CNNs for text** → Used for local patterns.
- **Transformers (BERT, GPT, T5, etc.)** → Current state-of-the-art.

🔷 Applications of NLP

✅ Google Translate (machine translation)

✅ ChatGPT (conversational AI)

✅ Siri/Alexa (speech-to-text + NLP)

✅ Spam filtering (email classification)

✅ Legal/medical document summarization

NLP Layers (Basic to Advanced)

1. Embedding Layer

- Converts words/tokens into **dense vectors** (instead of one-hot).
- Example: "cat" → [0.12, -0.87, 0.33, ...]
- Learns **semantic meaning**: similar words get similar vectors.
- Examples: Word2Vec, GloVe, BERT embeddings.

2. Recurrent Layers (RNN / LSTM / GRU)

- Handle **sequences** (text is sequential).
- **RNN**: passes hidden state through time.
- **LSTM / GRU**: special RNNs that solve vanishing gradient → remember long-term dependencies.
- Example: Processing sentence word-by-word.

3. Convolutional Layers (Text CNNs)

- Can be used for **N-gram feature extraction** in text.
- Kernel = sliding window over words.
- Example: Detecting phrases like “not good” (negative sentiment).

4. Attention Layer

- Core idea: Instead of encoding a whole sequence into one hidden state, allow the model to **focus on important words**.
- Example: In “The cat sat on the mat,” for predicting “mat,” attention focuses on “sat” and “on.”

5. Transformer Layers

- Built from:
  - **Multi-Head Self-Attention** (captures relationships between all words).
  - **Feedforward Networks** (dense layers applied to each position).
  - **Layer Normalization + Residual connections** (stabilize training).
- Stacked many times (e.g., 12, 24, 96 layers in big models).
- Used in BERT, GPT, T5, etc.

---

6. Position Encoding Layer

- Since Transformers don't have recurrence, they need **positional encodings** to know the order of words.
- Adds sine/cosine patterns or learned embeddings for word positions.

---

7. Output Layers

- Depends on task:
  - **Softmax layer** → classification (e.g., sentiment, next word prediction).
  - **Linear + Sigmoid** → multi-label classification.
  - **Decoder layers** → for text generation (translation, summarization).

🔷 Putting it together (Example: Transformer for NLP)

1. Input: "I love NLP"
2. **Embedding Layer** → turns words into vectors
3. **Positional Encoding** → adds word order
4. **Transformer Encoder Layers** → multi-head self-attention + feedforward
5. **Transformer Decoder Layers** (if generation task)
6. **Output Layer** → predicts class or next word

🔷 Summary

- **NLP layers** = the special building blocks that handle text data.
- Key ones:
  - Embedding Layer
  - RNN / LSTM / GRU
  - CNN for text
  - Attention & Transformer Layers
  - Output Layer

Linguistics from an NLP Perspective

Linguistics is the **scientific study of language**. NLP borrows heavily from its subfields. We can classify them like this:

---

1. Phonetics & Phonology (Sound Level)

- **Phonetics**: study of the sounds of human speech.
  - NLP connection → **Speech recognition, TTS (Text-to-Speech), phoneme modeling**.
- **Phonology**: how sounds function within a language (rules, patterns).
  - NLP connection → **accent detection, pronunciation modeling, speech synthesis**.

---

2. Morphology (Word Level)

🔷 Concept: Morphology

Morphology is the study of the **internal structure of words** — how words are formed from smaller meaningful units called **morphemes**.

👉 A **morpheme** is the smallest meaningful unit of language.

- Study of word structure and morphemes.
- NLP connection →
  - Tokenization (splitting words/subwords)
  - Stemming, Lemmatization
  - Handling **morphologically rich languages** (Turkish, Finnish, etc.).
  - Subword tokenizers (BPE, WordPiece, SentencePiece).

🔷 Types of Morphemes

1. **Free Morphemes**
   - Can stand alone as words.
   - Examples: "book," "cat," "happy."
2. **Bound Morphemes**
   - Cannot stand alone, must attach to another word.
   - Examples:
     - **Prefixes:** un- (unhappy), re- (rewrite).
     - **Suffixes:** -s (cats), -ed (played), -ing (running).

🔷 Morphology in NLP

In NLP, morphology helps with:

1. **Tokenization**
   - Splitting text into words/subwords.
   - Example: "playing" → ["play", "ing"].
2. **Stemming**
   - Reducing words to their root form (rule-based, crude).
   - Example: "running" → "run", "flies" → "fli".
3. **Lemmatization**

- More linguistically informed → reduces words to **dictionary root form**.
- Example: "running" → "run", "better" → "good".

4. **Handling Morphologically Rich Languages**
   - Some languages (e.g., Hindi, Turkish, Finnish) have many word forms due to inflections.
   - Example in Turkish: "ev" = house, "evlerimde" = in my houses (one word packs plural + possessive + locative case).
   - NLP models must break these into morphemes.

🔷 Why Morphology Matters in NLP

- Improves vocabulary handling (reduces sparsity).
- Helps in information retrieval and search engines.
- Essential in machine translation, especially for morphologically rich languages.
- Modern subword tokenizers (like **Byte Pair Encoding (BPE)**, WordPiece) are inspired by morphological analysis.

🔷 Morphological Typology

Morphological Typology is a classification system that categorizes languages based on how they form words and use morphemes (the smallest meaningful units). It distinguishes languages by their word structure patterns, such as whether they tend to combine multiple morphemes into single words or keep words as distinct units.

🔷 Summary

- **Morphology = study of word structure & morphemes.**
- **Morphemes = the smallest meaning units.**
- In NLP, morphology → tokenization, stemming, lemmatization, handling word variations.

---

3. Syntax (Sentence Structure)

- Study of how words combine to form **phrases & sentences**.
- NLP connection →
  - Parsing (constituency, dependency trees)
  - Grammar checkers
  - Machine Translation
  - Question Answering (QA)
  - Transformers model some aspects of syntax implicitly.

Example:

- Sentence: "The cat chased the mouse."
- Syntax tree helps NLP understand subject = "cat", object = "mouse".

---

4. Semantics (Meaning of Words & Sentences)

- Study of **meaning** in language.
- NLP connection →
  - Word embeddings (Word2Vec, GloVe, BERT embeddings).
  - Named Entity Recognition (NER).
  - Semantic similarity, entailment.
  - Knowledge graphs & semantic search.

Example:

- "Doctor" ≈ "Physician" (semantic similarity).

---

5. Pragmatics (Meaning in Context)

- How context influences meaning.
- NLP connection →
  - Conversational AI (ChatGPT 😉).

- Sarcasm detection.
- Dialogue systems (chatbots, assistants).
- Coreference resolution (who “he” refers to in a paragraph).

Example:

- "Can you open the window?" (literally a question, pragmatically a request).

---

6. Discourse (Beyond Sentences)

- Study of how multiple sentences connect to form coherent text.
- NLP connection →
  - Text summarization.
  - Document-level machine translation.
  - Dialogue modeling (keeping track of conversation flow).
  - Anaphora resolution (linking pronouns: "John went home. He was tired.").

---

7. Sociolinguistics & Psycholinguistics (Human Factors)

- **Sociolinguistics** → variation across regions, social groups, dialects.
  - NLP: multilingual models, dialect identification.
- **Psycholinguistics** → how humans process language in the brain.
  - NLP: cognitive-inspired models, interpretability.

🔷 How These Map to NLP Tasks

Linguistics Level	NLP Topics / Tasks
Phonetics/Phonology	Speech recognition, TTS
Morphology	Tokenization, stemming, lemmatization
Syntax	Parsing, grammar correction, translation

Linguistics Level	NLP Topics / Tasks
Semantics	Embeddings, QA, semantic search
Pragmatics	Dialogue systems, sentiment, sarcasm
Discourse	Summarization, coreference, long-text QA
Sociolinguistics	Multilingual NLP, bias/fairness
Psycholinguistics	Human-like AI, interpretability

🔷 Summary

- Linguistics gives the **levels of language**:

  Sounds → Words → Sentences → Meaning → Context → Discourse.
- NLP builds **models and layers** to handle each of these.
- Modern models like **Transformers (BERT, GPT, T5, etc.)** blur these distinctions, but they still rely on linguistic concepts.

🌐 Linguistics → NLP Mapping

1. Phonetics & Phonology (Sounds)
   └── NLP: Speech recognition, Text-to-Speech (TTS), pronunciation modeling
   └── Techniques: MFCCs, Spectrograms, RNNs, CNNs, Transformers (Wav2Vec2.0)
2. Morphology (Word structure, morphemes)
   └── NLP: Tokenization, Stemming, Lemmatization, Subword modeling
   └── Techniques: Rule-based stemmers, WordPiece, BPE, SentencePiece
3. Syntax (Sentence structure, grammar)
   └── NLP: Parsing, Grammar correction, Machine Translation
   └── Techniques: Dependency parsers, Constituency parsers, RNNs, Transformers
4. Semantics (Meaning of words/sentences)
   └── NLP: Word embeddings, Semantic similarity, NER, QA

└── Techniques: Word2Vec, GloVe, BERT, GPT embeddings, Knowledge graphs

5. Pragmatics (Meaning in context, intent)
   └── NLP: Dialogue systems, Sarcasm detection, Coreference resolution
   └── Techniques: Contextual embeddings, Transformers, Dialogue managers
6. Discourse (Beyond single sentences, text flow)
   └── NLP: Summarization, Document-level MT, Long-text QA
   └── Techniques: Transformers with attention (BERT, Longformer, GPT), Hierarchical models
7. Sociolinguistics (Language variation, dialects)
   └── NLP: Multilingual models, Bias/fairness, Dialect ID
   └── Techniques: mBERT, XLM-R, Large multilingual Transformers
8. Psycholinguistics (Human processing of language)
   └── NLP: Cognitive-inspired models, Interpretability, Human-like AI
   └── Techniques: Attention analysis, Human-in-the-loop learning

---