MODULE 6: Applications of NLP

Q.1) Text Summarization

Definition

Text summarization is the process of creating a shorter version of a text document while preserving its main meaning, ideas, and important information. It reduces large volumes of information into concise and coherent summaries, making it easier for users to understand the content quickly.

Types of Text Summarization

1. **Extractive Summarization**
    - Selects the most important sentences, phrases, or sections directly from the original text.
    - Works like highlighting key parts without rephrasing.
    - Example: Picking 5 key sentences from a news article.
2. **Abstractive Summarization**
    - Generates new sentences that capture the meaning of the text.
    - Similar to how a human would write a summary in their own words.
    - Example: Original → *"The stock market saw a steep fall yesterday due to global recession fears."*

      Abstractive Summary → *"Global recession concerns caused a market decline."*

Steps in Text Summarization

1. **Text Preprocessing**
    - Tokenization, stop word removal, stemming/lemmatization.
    - Example: Breaking text into words/sentences for easier processing.

2. **Feature Extraction / Representation**
    - Representing words/sentences using techniques like TF-IDF, Word Embeddings, or BERT embeddings.
3. **Sentence Scoring / Selection** (for extractive methods)
    - Assigning importance scores to sentences.
    - Selecting top-ranked ones for the summary.
4. **Summary Generation**
    - For extractive: Selected sentences are combined.
    - For abstractive: New text is generated using deep learning models.

Example

Original text (short version for demonstration):

"Artificial Intelligence is transforming industries. It helps businesses automate tasks, improve decision-making, and enhance customer experiences. However, challenges such as data privacy and job displacement must be addressed."

- **Extractive Summary:**

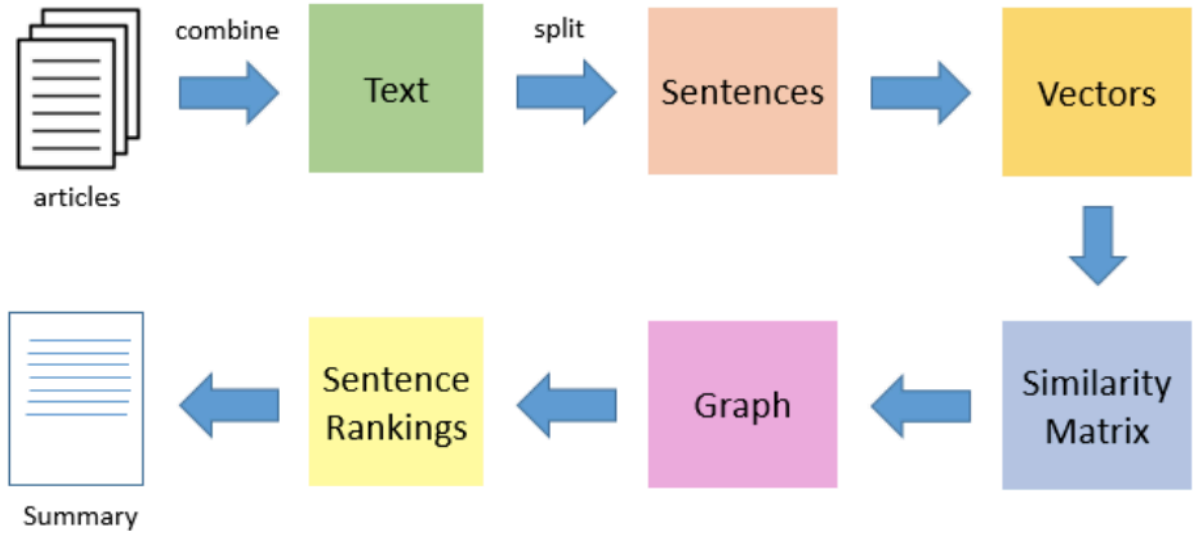
  "Artificial Intelligence is transforming industries. It helps businesses automate tasks and improve decision-making."

- **Abstractive Summary:**

  "AI is revolutionizing industries by automating work and aiding decisions, though it raises concerns like privacy and job loss."

Diagram (Process of Text Summarization)

Here's a conceptual diagram showing the summarization workflow:



Q.2) Steps in Text Processing for Information Retrieval (IR)

Information Retrieval (IR) deals with searching and retrieving relevant documents from large collections. Proper text processing is crucial before indexing and searching.

Steps in Text Processing for IR

1. **Tokenization**
   - Splitting text into smaller units (tokens) such as words, terms, or sentences.
   - Example: *"Information Retrieval is important"* → [Information, Retrieval, is, important]
2. **Stop Word Removal**
   - Removing common words that don't add much meaning (e.g., *is, the, of, and*).
   - Helps reduce noise.
3. **Normalization**
   - Converting text to a standard form.
   - Includes:
     - Lowercasing (e.g., *AI = ai*)

    - Removing punctuation/numbers.
    - Handling accents (*café → cafe*).
4. **Stemming / Lemmatization**
    - **Stemming:** Reducing words to their root form (e.g., *running → run*).
    - **Lemmatization:** Using vocabulary/grammar to get proper base form (e.g., *better → good*).
5. **Term Weighting (TF-IDF)**
    - Assigning weights to words based on importance.
    - Example: Rare words get higher importance than frequent ones.
6. **Indexing**
    - Creating inverted indexes for fast retrieval.
    - Example: Word → list of documents containing that word.
7. **Query Processing**
    - Preprocessing user's query the same way as documents.
    - Ensures fair matching.
8. **Ranking & Retrieval**
    - Using similarity measures (Cosine Similarity, BM25, etc.) to rank documents.
    - Returning the most relevant results.

✅ In short:

- **Text Summarization** condenses text into key ideas (extractive/abstractive).
- **Text Processing in IR** ensures text is cleaned, normalized, and structured for efficient searching.

Q.3) Rule-Based Machine Translation (RBMT) Systems

Definition

Rule-Based Machine Translation (RBMT) is one of the earliest approaches to Machine Translation. It relies on **linguistic rules** (morphology, syntax, semantics) and **bilingual dictionaries** to translate text from a source language to a target language.

Key Characteristics of RBMT

- Based on **linguistic knowledge** rather than statistical learning.
- Uses **grammatical analysis** of both source and target languages.
- Works best for **structured sentences** and grammatically rich languages.
- Requires **human experts** (linguists) to define grammar rules and dictionaries.

Main Components of RBMT

1. **Morphological Analysis**
    - Breaking down words into root + affixes.
    - Example: *"playing" → play + ing*.
2. **Syntactic Analysis**
    - Parsing sentences into grammatical structure (subject, verb, object).
    - Example: *"He eats an apple" → Subject = He, Verb = eats, Object = apple*.
3. **Transfer Rules (Mapping Source → Target)**
    - Converting source grammar structure into target grammar structure.
    - Example: English (SVO order: Subject-Verb-Object) → Japanese (SOV order: Subject-Object-Verb).
4. **Lexical Transfer (Dictionary Mapping)**
    - Mapping words from source dictionary to target dictionary.
    - Example: *House → Casa (Spanish)*.
5. **Generation in Target Language**
    - Producing grammatically correct output in the target language using rules.

Types of RBMT

1. **Direct Translation**
    - Word-for-word translation using bilingual dictionary.
    - Example: *"I am hungry"* → *"Yo soy hambre"* (incorrect grammar).
2. **Transfer-Based Translation**
    - Includes **syntactic + semantic transformations** before generating target.
    - More accurate than direct translation.
3. **Interlingua-Based Translation**
    - Converts source text into an **abstract meaning representation (interlingua)**.
    - Then generates target text from that interlingua.
    - Language-independent but requires very complex design.

Advantages of RBMT

- Does not require massive training data.
- Transparent and interpretable (rules are human-readable).
- Good for **specialized domains** and controlled language.

Limitations of RBMT

- Costly and time-consuming (requires linguistic experts).
- Difficult to scale across many languages.
- Produces rigid translations, sometimes lacking fluency.

Example

English: *"He is eating an apple."*

- Morphological Analysis: *He (PRON), is eating (Verb), an apple (NOUN PHRASE)*.
- Transfer Rules: English (SVO) → French (SVO).
- Output: *"Il mange une pomme."*

Q.4) Demonstrate the Working of Machine Translation Systems

To understand how **Machine Translation (MT) systems** work, let's look at the **general workflow** (applies to RBMT, SMT, and NMT with differences).

General Steps in Machine Translation

1. **Input Sentence (Source Language)**
   - User provides a sentence in one language.
   - Example: English → *"I am learning machine translation."*
2. **Preprocessing**
   - Tokenization (split into words/tokens).
   - Normalization (lowercasing, punctuation handling).
   - Morphological analysis (breaking words into root + suffixes).
3. **Source Language Analysis**
   - Identify grammatical structure, parts of speech, and meaning.
   - Example: *I (Subject), am learning (Verb), machine translation (Object).*
4. **Translation Method** (depends on approach):
   - **RBMT:** Uses rules and dictionaries.
   - **SMT:** Uses statistical probabilities.
   - **NMT:** Uses neural networks (encoder-decoder + attention).
5. **Transfer to Target Language**
   - Reordering according to target grammar rules.
   - Mapping words/phrases into target language equivalents.
6. **Target Sentence Generation**
   - Produces fluent, grammatically correct target sentence.
7. **Post-Processing**
   - Handle punctuation, capitalization, formatting.

Example Demonstration

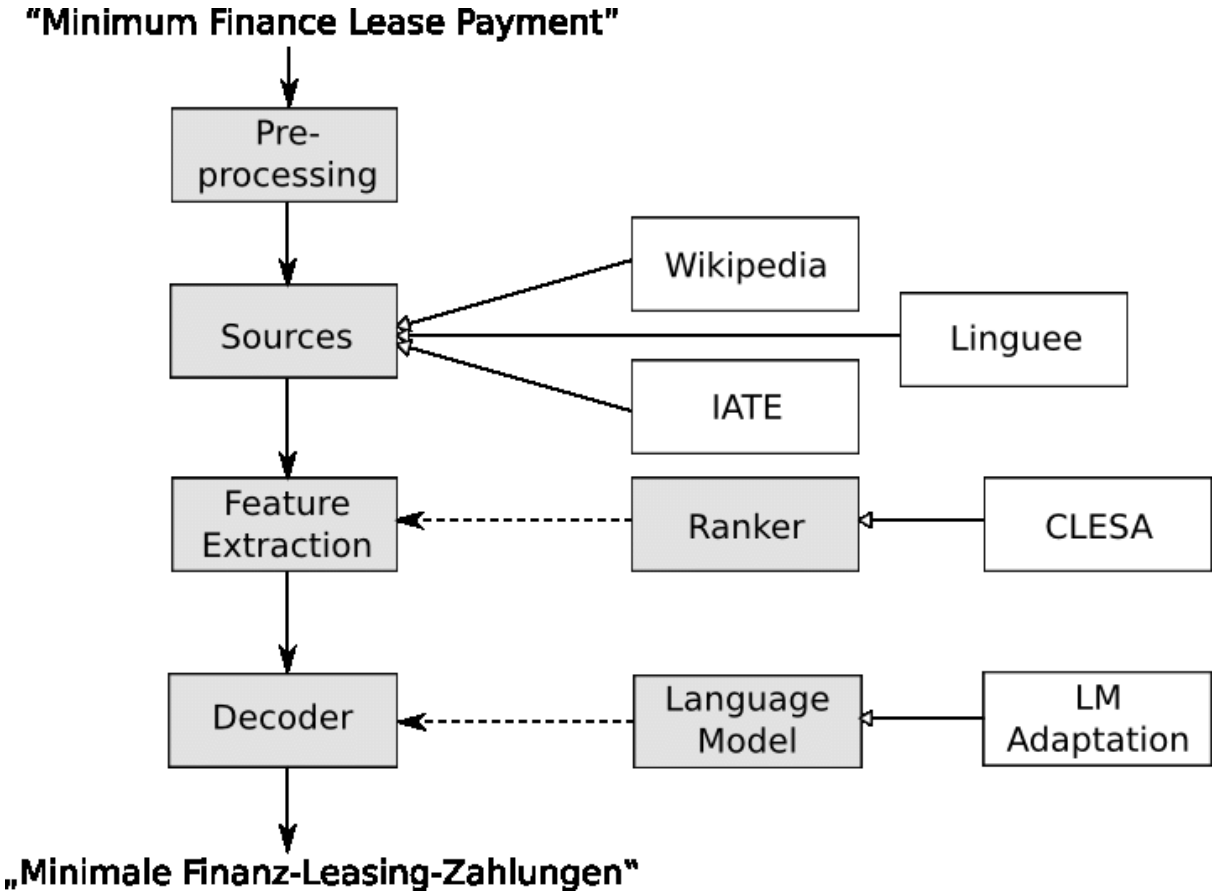
Sentence: “I am going to school.” (English → Spanish)

- **RBMT Approach:**
  - Dictionary lookup: *I → Yo, am going → voy, to school → a la escuela*
  - Output: *“Yo voy a la escuela.”*
- **SMT Approach:**
  - Probabilistic alignment: “I → Yo”, “going to school → voy a la escuela”
  - Output: *“Yo voy a la escuela.”*
- **NMT Approach:**
  - Encoder-decoder reads sentence meaning as vector representation.
  - Attention highlights important words.
  - Output: *“Voy a la escuela.”* (more natural, drops “Yo” since it’s implicit in Spanish).

Advantages of MT Systems

- Saves time in multilingual communication.
- Useful for global businesses, education, government, and healthcare.
- Improves continuously with data (especially NMT).

Diagram (Working of Machine Translation System)



Q.5) Machine Translation Approaches Used in NLP

Machine Translation (MT) is a core NLP task with several approaches. These have evolved from **rule-based methods** to **deep learning models**.

1. Rule-Based Machine Translation (RBMT)

- **Definition:** Uses linguistic rules (syntax, morphology, semantics) for translation.
- **Process:**
  1. Analyze source sentence grammatically.
  2. Map words/phrases using bilingual dictionaries.
  3. Reconstruct into target sentence.
- **Advantages:** Good for grammatically rich languages, interpretable.

- **Disadvantages:** Rule creation is costly and language-specific.

Example: *English: "He is eating an apple." → French: "Il mange une pomme."*

---

2. Statistical Machine Translation (SMT)

- **Definition:** Uses statistical models (language model + translation model).
- **Strengths:** Learns from large corpora, improves with more data.
- **Weaknesses:** Literal, weak handling of context, requires big parallel data.

---

3. Example-Based Machine Translation (EBMT)

- **Definition:** Relies on previously seen examples of translations (parallel corpora).
- **Process:** Matches input with similar past sentences, reuses translations.
- **Strengths:** Works well for fixed patterns.
- **Weaknesses:** Poor for unseen structures or sentences.

---

4. Hybrid Machine Translation (HMT)

- **Definition:** Combines rule-based + statistical + example-based methods.
- **Strengths:** Improves fluency and adequacy.
- **Weaknesses:** Complex to design and maintain.

---

5. Neural Machine Translation (NMT)

- **Definition:** Uses deep learning (neural networks) for end-to-end translation.
- **Architecture:**
  - Encoder (processes source sentence)
  - Decoder (generates target translation)
  - Attention Mechanism (focuses on important words during translation)
- **Advantages:**
  - Produces fluent, natural translations.
  - Handles long-distance dependencies better.

    - Learns contextual meaning.
- **Disadvantages:**
    - Requires huge training data and compute.
    - May hallucinate (produce fluent but wrong sentences).

Example: Google Translate (modern version) uses Transformer-based NMT.

Comparison of Approaches

Approach	Strengths	Weaknesses
RBMT	Precise grammar-based rules, interpretable	High cost of rule writing, poor scalability
SMT	Data-driven, scalable	Needs large corpora, less fluent
EBMT	Good for fixed patterns	Weak generalization
HMT	Combines strengths of methods	Complex system
NMT	Fluent, context-aware, state-of-art	Expensive, data-hungry

---

Q.6) Statistical Approach for Machine Translation

Definition

Statistical Machine Translation (SMT) is an approach to automatic translation that relies on statistical models derived from analyzing large amounts of bilingual text data. Instead of using explicit linguistic rules, SMT systems learn patterns of word/phrase correspondences between source and target languages.

Key Idea

The translation process is modeled as a **probability problem**:

$$T = \arg\max_{T} P(T|S)$$

Where:

- S = Source sentence (e.g., English)
- T = Target sentence (e.g., French)
- Goal: Find the translation T that maximizes the conditional probability given S.

Using **Bayes' Rule**:

$$P(T|S) = \frac{P(S|T) \cdot P(T)}{P(S)}$$

- $P(T)$ → **Language Model** (fluency in target language)
- $P(S|T)$ → **Translation Model** (how well source maps to target)

Main Components of SMT

1. **Language Model (LM):**
    - Ensures target output is grammatically correct and fluent.
    - Built from monolingual text in target language.
    - Example: Probability of word sequences like *"the cat is sleeping"*.
2. **Translation Model (TM):**
    - Learns word/phrase alignments between source and target.
    - Example: "house" ↔ "casa" (English ↔ Spanish).
3. **Decoder:**
    - Uses LM + TM to generate the most probable translation.
    - Example: If "I am hungry" is translated, decoder evaluates possible word orders in target language and selects best match.

Types of SMT Models

1. **Word-based Models**
    - Translate one word at a time.

- Limitation: Ignores multi-word expressions.

2. **Phrase-based Models**
   - Translate groups of words (phrases).
   - More accurate than word-based.
3. **Hierarchical Phrase-based Models**
   - Uses phrases and syntactic structures.
   - Captures long-distance dependencies better.
4. **Syntax-based SMT**
   - Uses parse trees and grammar rules to guide translation.

Advantages of SMT

- Data-driven: Requires no manual rules.
- Flexible: Works with any language pair given enough data.
- Scalable: Improves with more bilingual corpora.

Limitations

- Requires large parallel corpora.
- Struggles with rare words or idioms.
- Produces translations that are sometimes literal and lack deep semantic understanding.

Example

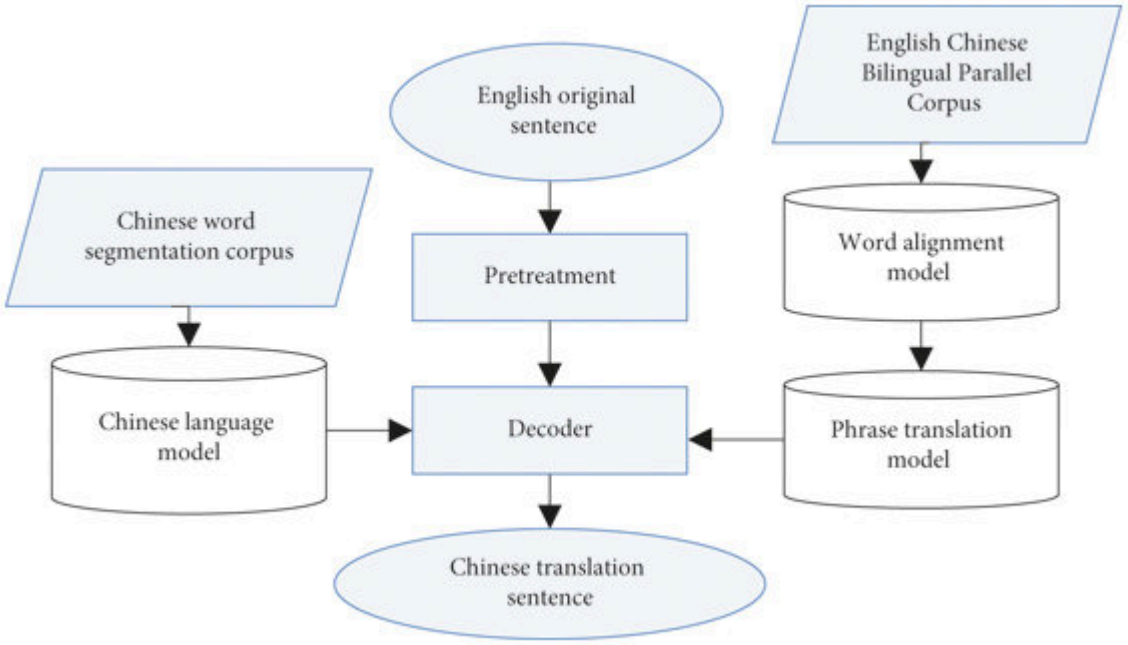
English: *"I am going to school."*

Spanish Translation by SMT: *"Yo voy a la escuela."*

Here:

- Word alignment: *I → Yo*, *am going → voy*, *to school → a la escuela*
- Language model ensures Spanish word order is correct.

Diagram (SMT Process):


English Chinese
Bilingual Parallel
Corpus
English original
sentence
Chinese word
segmentation corpus
Word alignment
model
Pretreatment
Chinese language
model
Decoder
Phrase translation
model
Chinese translation
sentence