SRM INSTITUTE OF SCIENCE & TECHNOLOGY

DEPARTMENT OF COMPUTING TECHNOLOGIES

21CSE356T

NATURAL LANGUAGE PROCESSING

UNIT- 5

Dr. S.PRABU
Assistant Professor
C-TECH-SRM-IST-KTR

UNIT 5

NLP APPLICATIONS

Introduction to Chatbot Applications, Retrieval based- Conversation based, Information Extraction and its approaches, Information Retrieval, Semantic Search and Evaluation, Question Answering, Summarization, Extractive Vs Abstractive Summarization, Machine Translation.

1. Introduction to Chatbot Applications

1. What is a Chatbot?

A chatbot is an artificial intelligence (AI) software designed to simulate conversation with human users, especially over the Internet. Chatbots are commonly used in customer service, education, healthcare, and e-commerce.

Chatbots can communicate via:

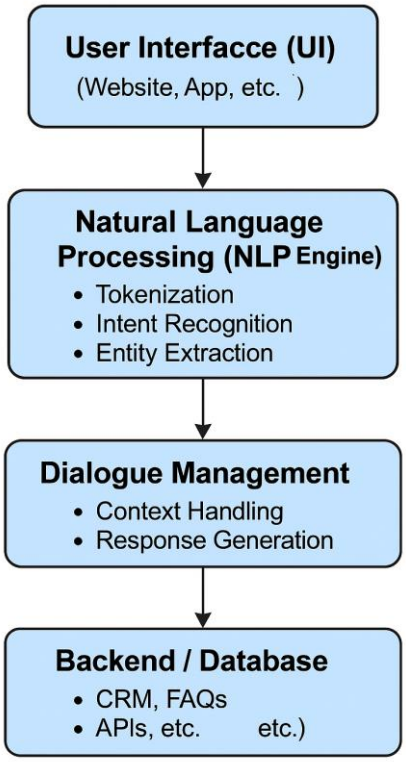
- Text-based interfaces (e.g., websites, messaging apps)
- Voice-based interfaces (e.g., Alexa, Google Assistant)

They rely heavily on Natural Language Processing (NLP) to interpret user input and generate appropriate responses.

2. Types of Chatbots

Type	Description
Rule-Based	Works on pre-defined rules or keywords. Limited in scope.
AI-Based	Uses NLP, machine learning, and contextual understanding for responses.
Hybrid	Combines rule-based logic with AI capabilities.

Components of a Chatbot System



Explanation of Chatbot Architecture Components

1. **User Interface (UI)**
    - Function: This is where users interact with the chatbot.
    - Examples: Websites, mobile apps, messaging platforms (like WhatsApp, Facebook Messenger).
    - Purpose: Captures user input (text or voice) and displays chatbot responses.
2. **Natural Language Processing (NLP Engine)**
    - Function: Analyzes the user input to understand meaning and intent.
    - Key Processes:
        - Tokenization: Splits the sentence into individual words or tokens.
        - Intent Recognition: Detects what the user wants (e.g., "book a ticket").
        - Entity Extraction: Identifies useful information like dates, locations, or names.

3. **Dialogue Management**
    - Function: Controls the conversation flow.
    - Tasks:
        - Context Handling: Keeps track of the conversation history and context.
        - Response Generation: Decides what the chatbot should reply based on intent and context. This can be rule-based or AI-based.
4. **Backend / Database**
    - Function: Provides data and services the chatbot needs to complete tasks.
    - Includes:
        - CRM (Customer Relationship Management): Stores user data or interaction history.
        - FAQs: Preloaded answers to common questions.
        - APIs: External services (e.g., flight booking, payment gateways) accessed by the chatbot.

4. NLP Techniques Used in Chatbots

- Tokenization: Splitting input text into words or phrases.
- Intent Recognition: Determining the goal of the user (e.g., booking, greeting).
- Entity Recognition: Extracting specific data (e.g., names, dates).
- Response Generation: Creating appropriate replies using templates or ML models.

5. Applications of Chatbots

Domain	Use Cases
Customer Support	24/7 support, order tracking, FAQ handling
Education	Virtual tutors, FAQs for students, language learning

Domain	Use Cases
Healthcare	Symptom checking, appointment booking, medication reminders
E-commerce	Product suggestions, order status, return policy inquiries
Banking	Balance inquiry, fraud alerts, account assistance

Challenges in Chatbot Development

- Handling ambiguous or unclear queries
- Context retention in long conversations
- Multilingual support
- Ensuring security and privacy

RETRIEVAL-BASED VS. CONVERSATION-BASED CHATBOTS

Chatbots can be broadly classified into two categories based on how they generate responses:

1. Retrieval-Based Chatbots

Definition:

Retrieval-based chatbots respond using predefined responses selected from a fixed set, based on the user's input.

How It Works:

- The bot uses intent classification and entity recognition to understand the input.
- Based on the intent, it retrieves the most appropriate response from a predefined list.
- It does not generate new sentences, only picks from existing ones.

Characteristics:

- Rule-based or ML-based classification
- Deterministic responses

- Easier to control behavior
- Doesn’t require large data for training

Example Use Cases:

- FAQs
- Customer service bots
- Banking support bots

Example:

User: "What is my account balance?"
Bot: "Your current balance is ₹5,000."

2. Conversation-Based Chatbots (Generative Chatbots)

Definition:

Conversation-based (or Generative) chatbots generate responses dynamically using deep learning models, typically sequence-to-sequence (seq2seq), transformers, or language models.

How It Works:

- It learns patterns in language through large conversational datasets.
- Uses models like RNNs, LSTMs, GPT, or BERT to generate responses word-by-word.
- Capable of handling open-domain conversations.

Characteristics:

- Requires large amounts of training data
- Context-aware and flexible
- Responses may be unpredictable
- Harder to control exact behavior

Example Use Cases:

- Personal assistants (like Siri, Alexa)
- Therapy/chat companion bots
- AI tutors or storytelling bots

Example:

User: "Tell me something interesting**."**
Bot: "Did you know honey never spoils? Archaeologists have found pots of honey in ancient Egyptian tombs that are over 3,000 years old!"

Comparison Table

Feature	**Retrieval-Based**	**Conversation-Based**
Response Type	Predefined	AI-generated (Dynamic)
Training Data Need	Low	High
Flexibility	Limited	High
Context Handling	Basic or none	Advanced (multi-turn conversation)
Accuracy	High for known queries	May vary depending on model performance
Use Case Scope	Narrow (domain-specific)	Broad (open-domain possible)

INFORMATION EXTRACTION AND ITS APPROACHES

1. What is Information Extraction (IE)?

Information Extraction (IE) is the process of automatically extracting structured information such as entities, relationships, and facts from unstructured or semi-structured text data.

IE transforms large volumes of natural language text into data that can be stored in a database or used for further analysis.

2. Key Tasks in Information Extraction

Task	Description
Named Entity Recognition (NER)	Identifies and classifies entities (e.g., names, dates, locations).
Relation Extraction	Detects relationships between entities (e.g., *works_at*, *born_in*).
Event Extraction	Finds and categorizes events described in the text.
Coreference Resolution	Identifies when different expressions refer to the same entity.

3. Example of Information Extraction

Input Sentence:
"Dr. A. P. J. Abdul Kalam was born in Rameswaram on October 15, 1931."

Extracted Information	Value
Person (Entity)	A. P. J. Abdul Kalam
Location (Entity)	Rameswaram
Date (Entity)	October 15, 1931
Relation	born_in(A. P. J., Rameswaram)

4. Approaches to Information Extraction

A. Rule-Based Approach

- Uses manually crafted rules and patterns (e.g., regular expressions).
- Example: "Mr. [Name]" → extracts person names using a fixed rule.
- Pros: Easy to implement for specific domains.
- Cons: Poor scalability and not adaptable to new patterns.

B. Statistical/Machine Learning Approach

- Uses supervised learning models like Decision Trees, CRFs (Conditional Random Fields), or SVMs trained on annotated datasets.
- Learns patterns from data rather than rules.
- Pros: Better generalization across domains.
- Cons: Requires large labeled training data and feature engineering.

C. Deep Learning Approach

- Uses models like BiLSTM, CNNs, and Transformers (e.g., BERT) to learn complex patterns from large datasets.
- Capable of handling context better.
- Pros: High accuracy, less manual feature extraction.
- Cons: Needs large datasets, high computational power.

D. Hybrid Approach

- Combines rule-based and machine learning/deep learning methods.
- Example: Use rules to pre-filter and then ML models for final prediction.
- Pros: Balance between accuracy and interpretability.
- Cons: Complexity increases in development.

5. Common NLP Models Used for IE

Model	Use in IE Tasks
spaCy	NER, POS tagging, dependency parsing
NLTK	Preprocessing and rule-based IE
BERT	Contextual NER and relationship extraction
Stanford NLP	Full pipeline for NER, Coreference, POS tagging

6. Applications of Information Extraction

- Academic Research: Extracting citations and author names
- Healthcare: Extracting symptoms and drug names from clinical notes
- E-commerce: Extracting product specs and customer reviews
- News Analytics: Extracting event details, people, and locations
- Business Intelligence: Extracting company information and financial data

INFORMATION RETRIEVAL (IR)

1. What is Information Retrieval?

Information Retrieval (IR) is the process of searching for relevant information from a large collection of unstructured data, such as documents, websites, or databases.

IR systems match a user's query with relevant documents or content using algorithms that analyze keywords, context, and semantics.

Example: When you type a query in Google, the system retrieves web pages that are most relevant to the input text — this is information retrieval.

2. Components of an IR System

Component	Description
Corpus	A collection of documents or texts to be searched.
Query	The user's search input.
Tokenizer/Parser	Breaks down the text into meaningful units (words or phrases).
Indexing Engine	Builds a searchable index from the documents (like a keyword map).
Retrieval Model	Scores and ranks documents based on how well they match the query.
Ranking Module	Orders the results by relevance score.
Results Display	Presents the top documents to the user.

3. Types of Information Retrieval

Boolean Retrieval

- Uses logical operators (AND, OR, NOT) to match documents.
- Example: "AI AND healthcare" retrieves documents that contain both terms.

Vector Space Model (VSM)

- Represents documents and queries as vectors in a multi-dimensional space.
- Uses cosine similarity to rank results.

Probabilistic Retrieval

- Calculates the probability that a document is relevant to the query.
- Includes models like BM25.

Neural IR / Deep Learning-Based Retrieval

- Uses embeddings (e.g., BERT, SBERT) to match semantic meanings, not just keywords.

4. Information Retrieval vs Information Extraction

Feature	Information Retrieval	Information Extraction
Goal	Find relevant documents	Extract structured data from text
Input	Query (user input)	Raw text or document
Output	Ranked documents	Entities, relations, structured output
Example	Search for "COVID vaccine stats"	Extract "Pfizer - 95% efficacy"

5. Evaluation Metrics in IR

Metric	Description
Precision	Proportion of retrieved documents that are relevant.
Recall	Proportion of relevant documents that were retrieved.
F1-Score	Harmonic mean of Precision and Recall.
Mean Average Precision (MAP)	Measures average precision for a set of queries.
NDCG (Normalized Discounted Cumulative Gain)	Takes the ranking order into account.

6. Applications of Information Retrieval

- Web Search Engines (e.g., Google, Bing)
- Document Management Systems
- Legal and Academic Research
- E-commerce Product Search
- Digital Libraries and Archives

SEMANTIC SEARCH AND EVALUATION

1. What is Semantic Search?

Semantic Search is an advanced type of information retrieval that focuses on the meaning and context of a query rather than just matching keywords. It tries to understand the intent behind the search and return results that are conceptually relevant, not just lexically similar.

2. How Semantic Search Works

Step	Description
1. Query Understanding	The system interprets the query's context, intent, and entities.
2. Text Embedding	Words, sentences, and documents are converted into vector representations (using models like BERT, SBERT).
3. Similarity Calculation	Computes semantic similarity (e.g., cosine similarity) between the query and documents.
4. Ranking & Retrieval	Documents are ranked by meaning relevance, not just keyword frequency.

Example Comparison

Query	Traditional IR Result	Semantic Search Result
"Best phone with good battery"	Documents with “best,” “phone,” “battery”	Phones ranked based on battery life reviews
"Symptoms of COVID-19"	Articles with matching terms	Content describing the disease in context

3. Techniques Used in Semantic Search

Technique	Description
Word Embeddings	Represents words in vector space (e.g., Word2Vec, GloVe).
Sentence Embeddings	Captures context at sentence level (e.g., Universal Sentence Encoder, SBERT).
Transformer Models	Understands contextual meaning (e.g., BERT, RoBERTa, T5).
Knowledge Graphs	Connects entities and their relationships for deeper understanding.

4. Advantages of Semantic Search

- Understands synonyms and paraphrased queries
- Handles longer and conversational queries
- Improves user experience with more relevant results
- Enables personalized recommendations

5. Evaluation of Semantic Search

To evaluate how well a semantic search engine performs, we use standard IR metrics, along with semantic-specific assessments:

Metric	Purpose
Precision@k	Measures how many of the top-k results are relevant.
Recall	Measures how much of the total relevant information was retrieved.
Mean Reciprocal Rank (MRR)	Evaluates how high the first relevant result appears in the ranking.
NDCG (Normalized Discounted Cumulative Gain)	Accounts for the relevance and position of results.
Semantic Similarity Score	Measures similarity between the query and retrieved text using embeddings.

6. Applications of Semantic Search

- AI-powered search engines (e.g., Google's BERT-enhanced search)
- E-commerce search: "Shoes for wedding" → returns formal footwear
- Academic research engines (e.g., Semantic Scholar)
- Voice assistants (e.g., Siri, Alexa)
- Enterprise knowledge management systems

QUESTION ANSWERING (QA)

1. What is Question Answering in NLP?

Question Answering (QA) is a task in Natural Language Processing that focuses on automatically answering a question posed in natural language. QA systems aim to return precise and relevant answers from structured or unstructured data sources, like documents, webpages, or knowledge bases.

Example:
Input: "Who is the president of India?"
Output: "Droupadi Murmu"

2. Types of QA Systems

Type	Description
Closed-domain QA	Answers questions from a specific domain (e.g., medical, legal).
Open-domain QA	Answers any general question using broad data like Wikipedia.
Factoid QA	Answers are short and factual (names, dates, etc.).
List QA	Returns a list of relevant items.
Yes/No QA	Produces binary answers.
Descriptive QA	Returns full-sentence or paragraph-level answers.

3. Components of a QA System

Component	Role
Question Processing	Analyzes and understands the question: intent, type, key entities.
Document Retrieval	Finds relevant documents/passages from a corpus (IR phase).
Passage Ranking	Ranks retrieved text chunks by relevance.
Answer Extraction	Extracts the most relevant span or generates a direct answer.

Component	Role
Answer Generation	(For generative QA) Forms the final answer using language models.

Example Process:

Question: “When was Google founded?”

1. Query Analysis → Looks for entities: “Google”, “founded”
2. Document Retrieval → Finds a Wikipedia article about Google
3. Answer Extraction → Extracts: “September 4, 1998”

4. QA Approaches

Information Retrieval-based QA (Extractive)

- Finds answers within retrieved documents.
- Uses models like BERT, RoBERTa.
- High accuracy for fact-based queries.

Generative QA

- Generates answers from scratch using a language model (e.g., T5, GPT).
- Useful for complex or descriptive questions.
- Handles multi-hop reasoning and summarization.

Knowledge Base QA (KB-QA)

- Uses structured databases like Wikidata or Freebase.
- Converts questions into SPARQL or SQL queries.
- Example: "What is the capital of France?" → Query to a geographical knowledge graph.

5. Evaluation Metrics for QA Systems

Metric	Description
Exact Match (EM)	Measures if the system's answer exactly matches the gold-standard answer.
F1 Score	Considers both precision and recall in partial matches.
BLEU / ROUGE	Used for evaluating generated answers.
Human Evaluation	Judges fluency, relevance, and completeness manually.

6. Applications of QA

- Chatbots and Virtual Assistants (Siri, Google Assistant)
- Academic QA engines (e.g., Elicit, SciQA)
- Customer Support Automation

  Enterprise Knowledge Access
- Legal/Healthcare Document Search and QA
- Voice-based search engines

SUMMARIZATION (EXTRACTIVE VS. ABSTRACTIVE)

1. What is Summarization in NLP?

Summarization is the process of generating a short, coherent, and meaningful summary of a larger text while preserving the most important content and intent.

There are two main approaches to summarization in NLP:

- Extractive Summarization
- Abstractive Summarization

2. Extractive Summarization

Definition:

Extractive summarization involves selecting key sentences or phrases from the original text and combining them to form a summary. It does not generate new words; it just picks the most important existing ones.

Techniques:

- TextRank (graph-based)
- TF-IDF scoring
- LexRank
- BERTSum (Transformer-based)

Advantages:

- Easier to implement
- Preserves original grammar and wording

Limitations:

- May produce less coherent or redundant summaries
- Can't paraphrase or combine information

Example:

Input Text:
"The Indian Space Research Organisation (ISRO) successfully launched Chandrayaan-3. It aims to land near the Moon's south pole and conduct experiments."

Extractive Summary:
"ISRO successfully launched Chandrayaan-3. It aims to land near the Moon's south pole."

3. Abstractive Summarization

Definition:

Abstractive summarization generates new sentences that convey the core ideas of the original text, similar to how a human might summarize.

Techniques:

- Seq2Seq models with attention
- Transformers (BART, T5, PEGASUS, GPT)
- Reinforcement learning for better fluency

Advantages:

- More human-like summaries
- Better fluency and paraphrasing
- Can remove redundancy

Limitations:

- Harder to train
- May generate inaccurate or hallucinated facts

Example:

Input Text:
"The Indian Space Research Organisation (ISRO) successfully launched Chandrayaan-3. It aims to land near the Moon's south pole and conduct experiments."

Abstractive Summary:
"ISRO's Chandrayaan-3 mission targets a lunar south pole landing for scientific research."

4. Comparison Table

Feature	Extractive Summarization	Abstractive Summarization
Approach	Selects sentences from original text	Rewrites text using natural language
Output Quality	Factual but may be choppy or redundant	More fluent and concise
Grammar Control	Relies on original text grammar	Can rephrase for better grammar
Training Complexity	Easier to train	Requires large datasets and models
Common Tools	TextRank, LexRank, BERTSum	BART, T5, GPT, PEGASUS

5. Evaluation Metrics

Metric	Use
ROUGE Score	Measures overlap between machine-generated and reference summaries (ROUGE-1, ROUGE-2, ROUGE-L).
BLEU Score	Measures n-gram precision, more common in machine translation.
METEOR	Considers synonymy and word order.
Human Evaluation	Judges fluency, relevance, coherence, and informativeness.

6. Applications of Summarization

- News article summarization
- Meeting and legal transcript summarization
- Chat and customer feedback summarization
- Scientific paper summarization
- Email and document summarization in business tools

MACHINE TRANSLATION (MT)

1. What is Machine Translation?

Machine Translation is a subfield of NLP that focuses on automatically translating text or speech from one language to another. It enables cross-lingual communication and plays a key role in global information access.

Example:
Input (English): "How are you?"
Output (French): "Comment ça va ?"

2. Types of Machine Translation Approaches

A. Rule-Based Machine Translation (RBMT)

- Based on linguistic rules, grammar, and bilingual dictionaries.
- Translates using syntax trees and morphological analysis.

Pros:

- Linguistically sound and interpretable

Cons:

- Requires manual rule creation for each language pair
- Not scalable to many languages

B. Statistical Machine Translation (SMT)

- Uses probabilities and parallel corpora to learn translations.
- Includes Phrase-Based and Word-Based SMT models.

Pros:

- Data-driven, requires less manual effort

Cons:

- May produce grammatically incorrect or unnatural translations
- Can't capture context well

C. Neural Machine Translation (NMT)

- Uses deep learning, especially sequence-to-sequence (Seq2Seq) models with attention mechanisms.
- Latest models use Transformers (e.g., Google's BERT, T5, OpenNMT, Facebook's fairseq).

Pros:

- Fluent, context-aware, and high-quality translations
- Supports low-resource languages with transfer learning

Cons:

- Requires large amounts of data and computation
- May generate hallucinated or biased outputs

3. Key Components of NMT Systems

Component	Role
Encoder	Converts the source sentence into a hidden representation
Decoder	Generates the translated sentence in the target language
Attention Mechanism	Focuses on relevant parts of the input while decoding each word
Transformer Architecture	Enables parallel processing and long-range dependency capture

Example Architecture:

- Input: "I love NLP."
- Encoder: Processes "I", "love", "NLP" → produces vector representations
- Attention: Highlights "love" when translating verb
- Decoder: Generates translated sentence → "J'adore le traitement du langage naturel."

4. Evaluation Metrics

Metric	Description
BLEU Score	Measures n-gram overlap between machine and human translations
METEOR	Includes synonym matching and stem variations
TER (Translation Edit Rate)	Measures number of edits needed to fix the machine translation
Human Evaluation	Rates fluency, adequacy, and faithfulness

5. Challenges in Machine Translation

- Handling idioms and metaphors
- Word order differences between languages
- Ambiguity and context sensitivity
- Low-resource languages with limited training data
- Maintaining domain-specific accuracy (e.g., legal vs. medical terms)

6. Applications of Machine Translation

- Multilingual websites and content translation (e.g., Google Translate)
- Mobile translation apps (e.g., Microsoft Translator, iTranslate)
- Academic paper and book translation
- Live speech translation (e.g., in video conferencing)
- Real-time chat translation in social media platforms