CCS369 TEXT AND SPEECH ANALYSIS

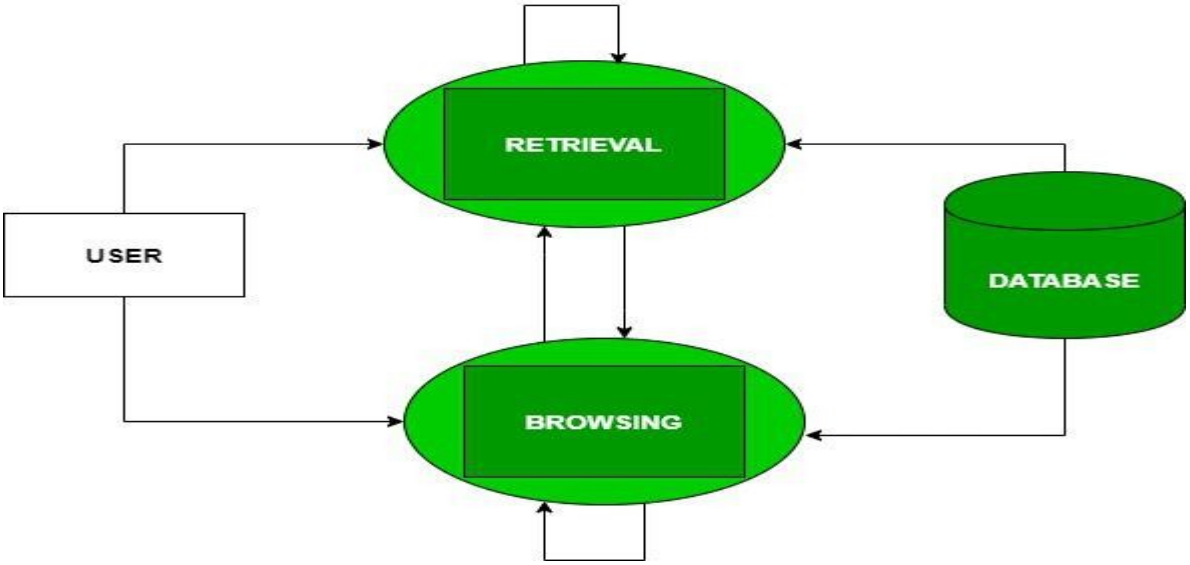
UNIT III QUESTION ANSWERING AND DIALOGUE SYSTEMS

Information retrieval:

What is text information retrieval?

• Text retrieval is to return relevant textual documents from a given collection, according to users' information needs as declared in a query.
• Main differences from database retrieval are concerned with: – Information.
• Unstructured text vs.

Information Retrieval (IR) can be defined as a software program that deals with the organization, storage, retrieval, and evaluation of information from document repositories, particularly textual information. Information Retrieval is the activity of obtaining material that can usually be documented on an unstructured nature **i.e.** usually text which satisfies an information need from within large collections which is stored on computers..

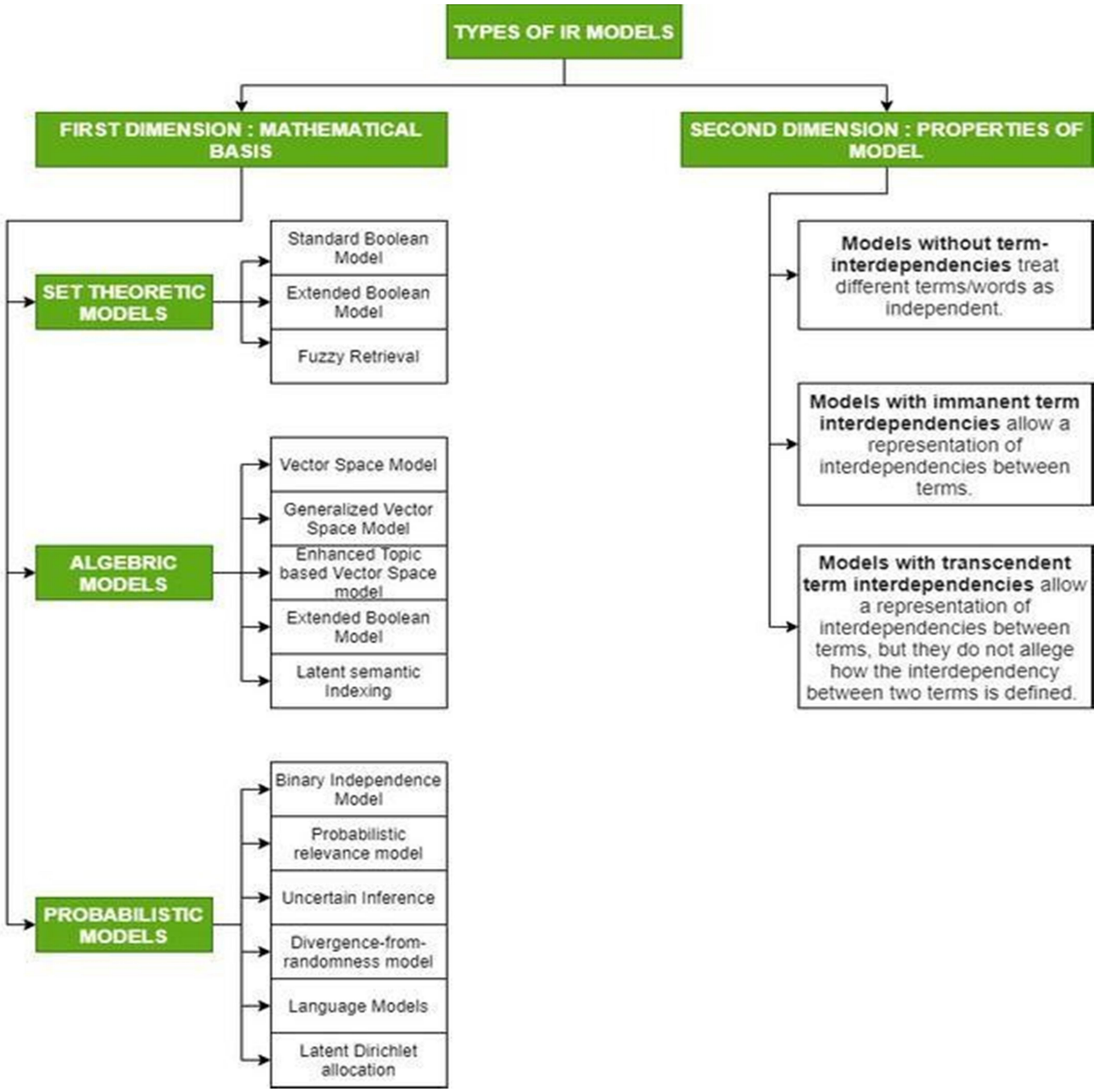


Examples:
Vector-space, Boolean and Probabilistic IR models. In this system, the retrieval of information depends on documents containing the defined set of queries.

What is an IR Model?

An Information Retrieval (IR) model selects and ranks the document that is required by the user or the user has asked for in the form of a query. The documents and the queries are represented in a similar manner, so that document selection and ranking can be formalized by a matching function that returns a **retrieval status value (RSV)** for each document in the collection. Many of the Information Retrieval systems represent document contents by a set of

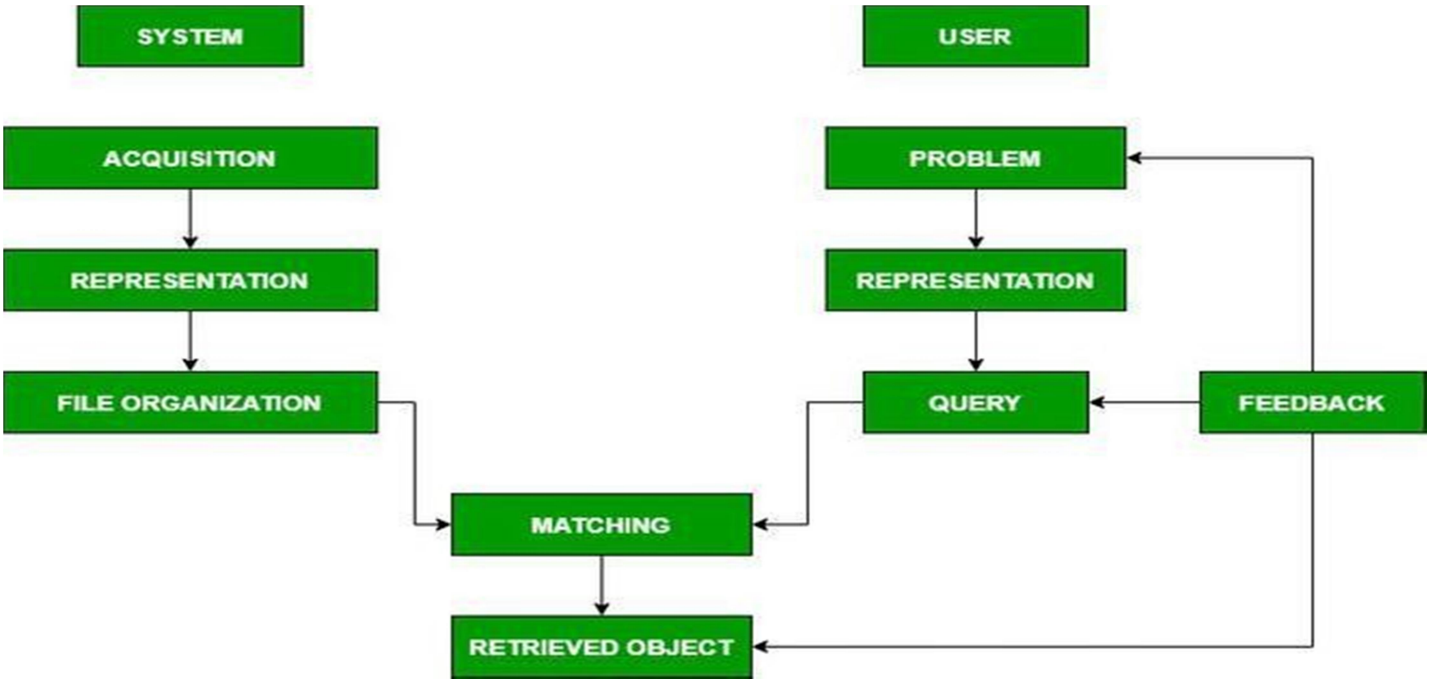
descriptors, called terms, belonging to a vocabulary V. An IR model determines the query-document matching function according to four main approaches:



Components of Information Retrieval/ IR Model

- **Acquisition:** In this step, the selection of documents and other objects from various web resources that consist of text-based documents takes place. The required data is collected by web crawlers and stored in the database.
  - **Representation:** It consists of indexing that contains free-text terms, controlled vocabulary, manual & automatic techniques as well. example: Abstracting contains

summarizing and Bibliographic description that contains author, title, sources, data, and metadata.



- **Representation:** It consists of indexing that contains free-text terms, controlled vocabulary, manual & automatic techniques as well. example: Abstracting contains summarizing and Bibliographic description that contains author, title, sources, data, and metadata.
- **File Organization:** There are two types of file organization methods. i.e. *Sequential*: It contains documents by document data. *Inverted*: It contains term by term, list of records under each term. *Combination* of both.
- **Query:** An IR process starts when a user enters a query into the system. Queries are formal statements of information needs, for example, search strings in web search engines. In information retrieval, a query does not uniquely identify a single object in the collection. Instead, several objects may match the query, perhaps with different degrees of relevancy.

Difference Between Information Retrieval and Data Retrieval

Information Retrieval	Data Retrieval
The software program that deals with the organization, storage, retrieval, and evaluation of information from document repositories particularly textual information.	Data retrieval deals with obtaining data from a database management system such as ODBMS. It is A process of identifying and retrieving the data from the database, based on the query provided by user or application.
Retrieves information about a subject.	Determines the keywords in the user query and

Information Retrieval	Data Retrieval
	retrieves the data.
Small errors are likely to go unnoticed.	A single error object means total failure.
Not always well structured and is semantically ambiguous.	Has a well-defined structure and semantics.
Does not provide a solution to the user of the database system.	Provides solutions to the user of the database system.
The results obtained are approximate matches.	The results obtained are exact matches.
Results are ordered by relevance.	Results are unordered by relevance.
It is a probabilistic model.	It is a deterministic model.

The User Task: The information first is supposed to be translated into a query by the user. In the information retrieval system, there is a set of words that convey the semantics of the information that is required whereas, in a data retrieval system, a query expression is used to convey the constraints which are satisfied by the objects.

- **Logical View of the Documents:** A long time ago, documents were represented through a set of index terms or keywords. Nowadays, modern computers represent documents by a full set of words which reduces the set of representative keywords. This can be done by eliminating stopwords i.e. articles and connectives. These operations are text operations. These text operations reduce the complexity of the document representation from full text to set of index terms.

Past, Present, and Future of Information Retrieval

1. Early Developments: As there was an increase in the need for a lot of information, it became necessary to build data structures to get faster access. The index is the data structure for faster retrieval of information. Over centuries manual categorization of hierarchies was done for indexes.

2. Information Retrieval In Libraries: Libraries were the first to adopt IR systems for information retrieval. In first-generation, it consisted, automation of previous technologies, and the search was based on author name and title. In the second generation, it included searching by subject heading, keywords, etc. In the third generation, it consisted of graphical interfaces, electronic forms, hypertext features, etc.

3. The Web and Digital Libraries: It is cheaper than various sources of information, it provides greater access to networks due to digital communication and it gives free access to publish on a larger medium.

Advantages of Information Retrieval

1. Efficient Access: Information retrieval techniques make it possible for users to easily locate and retrieve vast amounts of data or information.

2. Personalization of Results: User profiling and personalization techniques are used in information retrieval models to tailor search results to individual preferences and behaviors.

3. Scalability: Information retrieval models are capable of handling increasing data volumes.

4. Precision: These systems can provide highly accurate and relevant search results, reducing the likelihood of irrelevant information appearing in search results.

Disadvantages of Information Retrieval

1. Information Overload: When a lot of information is available, users often face information overload, making it difficult to find the most useful and relevant material.

2. Lack of Context: Information retrieval systems may fail to understand the context of a user's query, potentially leading to inaccurate results.

3. Privacy and Security Concerns: As information retrieval systems often access sensitive user data, they can raise privacy and security concerns.

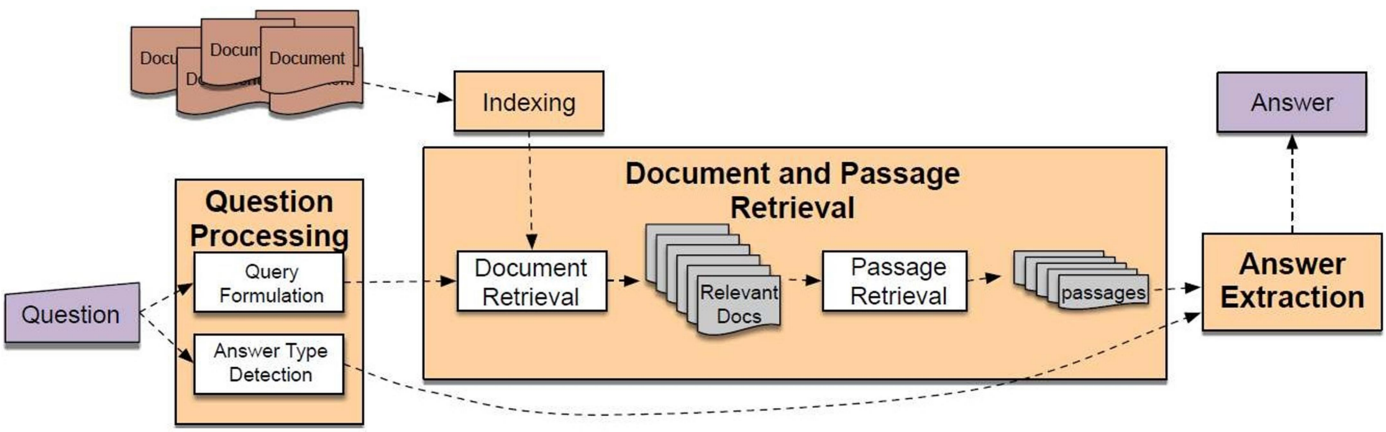
4. Maintenance Challenges: Keeping these systems up-to-date and effective requires ongoing efforts, including regular updates, data cleaning, and algorithm adjustments.

5. Bias and fairness: Ensuring that information retrieval systems do not exhibit biases and provide fair and unbiased results is a crucial challenge, especially in contexts like web search engines and recommendation systems.

IR-based question answering:

What is IR based question answering?

IR-based Factoid Question Answering. The goal of information retrieval based question answering is to answer a user's question by finding short text segments on the web or some other collection of doc- uments.



What is a question-answering System?

Question answering (QA) is a field of natural language processing (NLP) and artificial intelligence (AI) that aims to develop systems that can understand and answer questions posed in natural language.

How does a natural language question-answering system work?

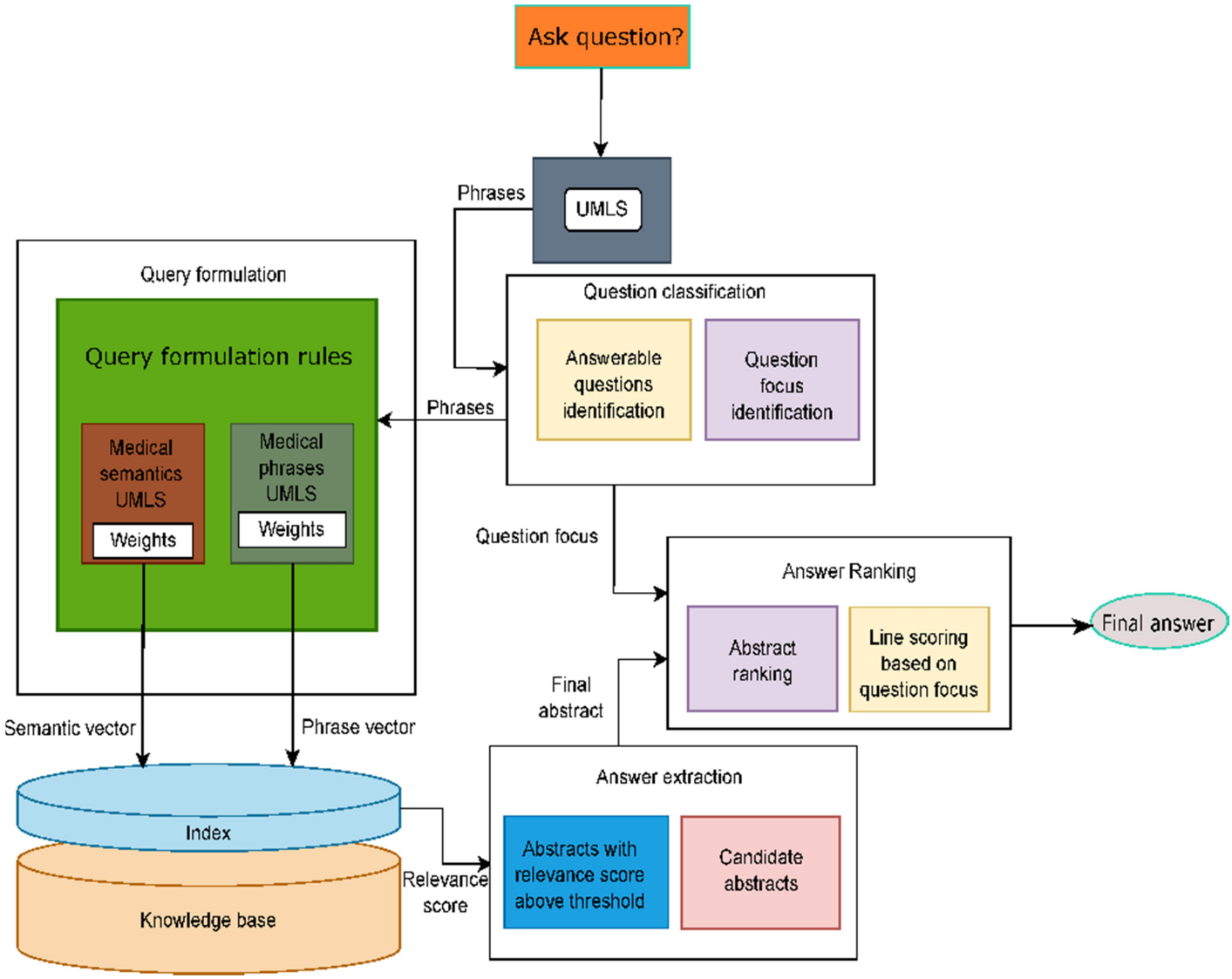
A natural language question-answering (QA) system is a computer program that automatically answers questions using NLP. The basic process of a natural language QA system includes the following steps:

1. **Text pre-processing**: The question is pre-processed to remove irrelevant information and standardise the text's format. This step includes tokenisation, lemmatisation, and stop-word removal, among others.
2. **Question understanding**: The pre-processed question is analysed to extract the relevant entities and concepts and to identify the type of question being asked. This step can be done using natural language processing (NLP) techniques such as named entity recognition, dependency parsing, and part-of-speech tagging.
3. **Information retrieval**: The question is used to search a database or corpus of text to retrieve the most relevant information. This can be done using information retrieval techniques such as keyword search or semantic search.
4. **Answer generation:** The retrieved information is analysed to extract the specific answer to the question. This can be done using various techniques, such as machine learning algorithms, rule-based systems, or a combination.
5. **Ranking**: The extracted answers are ranked based on relevance and confidence score.

Types of question answering system

1. Information retrieval-based QA

Information retrieval-based question answering (QA) is a method of automatically answering questions by searching for relevant documents or passages that contain the answer. This approach uses information retrieval techniques, such as keyword or semantic search, to identify the documents or passages most likely to hold the answer to a given question.



2. Knowledge-based QA

Knowledge-based question answering (QA) automatically answers questions using a knowledge base, such as a database or ontology, to retrieve the relevant information. This strategy's foundation is that searching for a structured knowledge base for a question can yield the answer.

Knowledge-based QA systems are generally more accurate and reliable than other QA approaches based on structured and well-curated knowledge.

3. Generative QA

Generative question answering (QA) automatically answers questions using a generative model, such as a neural network, to generate a natural language answer to a given question.

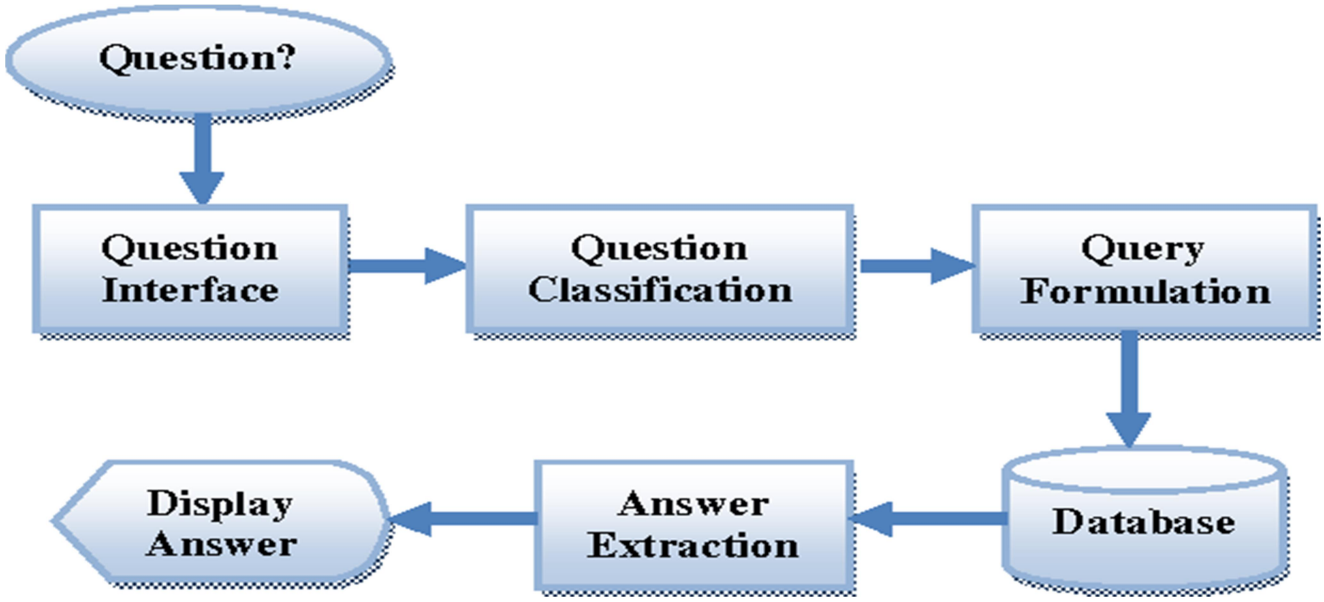
This method is based on the idea that a machine can be taught to understand and create text in natural language to provide a correct answer in terms of grammar and meaning.

4. Hybrid QA

Hybrid question answering (QA) automatically answers questions by combining multiple QA approaches, such as information retrieval-based, knowledge-based, and generative QA. This approach is based on the idea that different QA approaches have their strengths and weaknesses, and by combining them, the overall performance of the QA system can be improved.

5. Rule-based QA

Rule-based question answering (QA) automatically answers questions using a predefined set of rules based on keywords or patterns in the question. This approach is based on the idea that many questions can be answered by matching the question to a set of predefined rules or templates.



Applications:

- Customer Service
- Search engines
- Healthcare
- Education
- Finance
- In e-commerce
- Void assistants
- Chatbots
- Virtual assistants
- Business intelligence

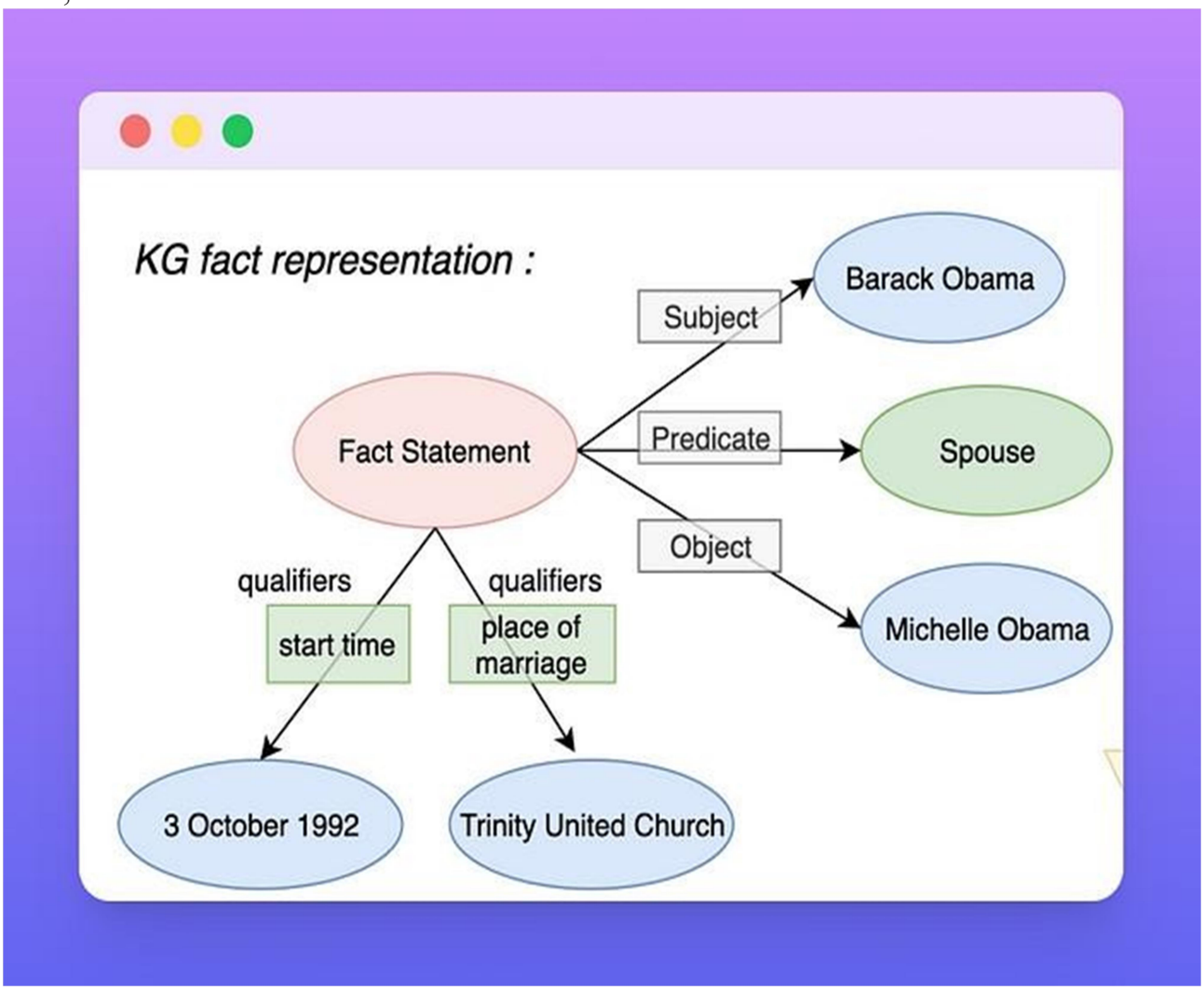
Tools:

TensorFlow
BERT
GPT-3
Hugging Face
SpaCy
NLTK
OpenNLP

knowledge-based question answering:

What is knowledge based question answering?

Knowledge-based question answering (KBQA) is the task of finding answers to questions by pro- cessing a structured knowledge base KB. A KB consists of a set of entities E, a set of relations R, and a set of literals S.



Knowledge-based question answering (KBQA) in text and speech analysis involves using structured knowledge bases or ontologies to answer questions posed in natural language. This approach contrasts with traditional information retrieval systems, which primarily match keywords or phrases to documents. Here's an overview of how KBQA works:

Knowledge Representation: KBQA systems rely on structured knowledge representations such as ontologies, knowledge graphs, or semantic networks. These representations capture entities, their attributes, relationships, and hierarchies in a formalized manner.

Natural Language Understanding: The system analyzes the natural language question to understand its meaning, including entity mentions, relationships, and constraints implied by the question. Techniques such as part-of-speech tagging, named entity recognition, dependency parsing, and semantic role labeling are often used.

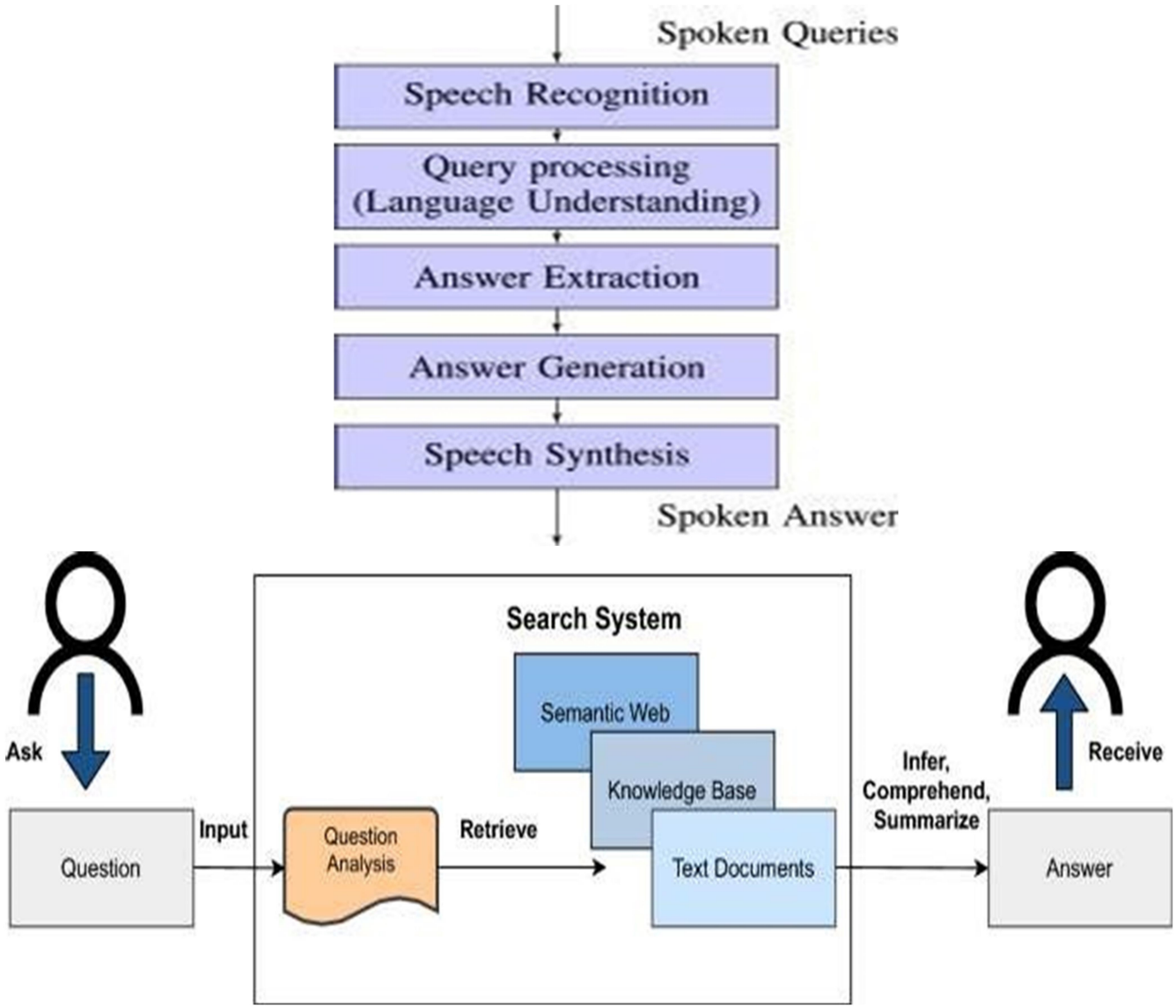
Query Formulation: Based on the understanding of the question, the system formulates a structured query that can be executed against the knowledge base. This query typically involves selecting relevant entities, properties, and relationships to retrieve the desired information.

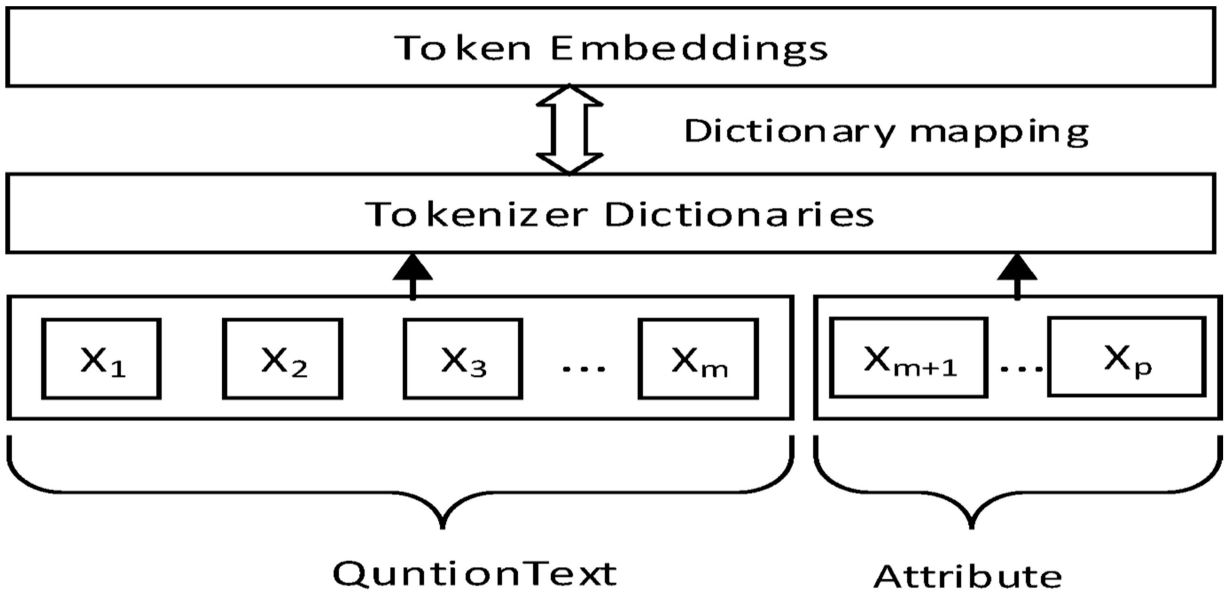
Knowledge Base Querying: The formulated query is executed against the knowledge base to retrieve relevant information. This process may involve querying a structured database, a knowledge graph, or accessing external sources such as linked data on the web.

Answer Generation: Once the relevant information is retrieved from the knowledge base, it is processed to generate a natural language answer that directly addresses the user's question. This may involve aggregating and summarizing information, as well as ensuring that the answer is fluent and grammatically correct.

Response Presentation: Finally, the generated answer is presented to the user through the appropriate interface, whether it's a text-based response in a chatbot interface or synthesized speech in a voice-based interaction.

KBQA systems can vary in complexity and sophistication, ranging from simple rule-based approaches to more advanced systems leveraging machine learning and natural language processing techniques.


Spoken Queries
Speech Recognition
Query processing
(Language Understanding)
Answer Extraction
Answer Generation
Speech Synthesis
Spoken Answer
Search System
Semantic Web
Knowledge Base
Text Documents
Ask
Question
Input
Question
Analysis
Retrieve
Infer,
Comprehend,
Summarize
Receive
Answer

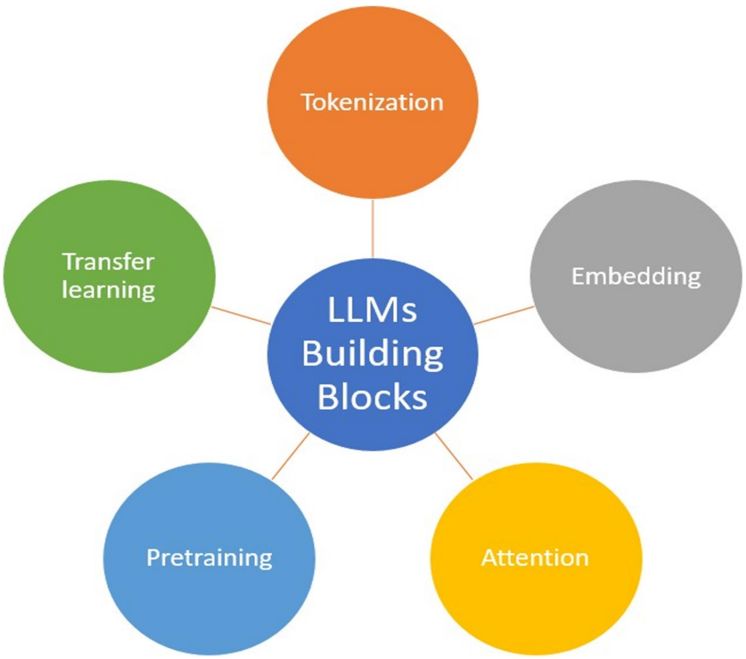

Token Embeddings
Dictionary mapping
Tokenizer Dictionaries
X_1
X_2
X_3
...
X_m
X_{m+1}
...
X_p
QuntionText
Attribute

language models for QA:

These models can predict any word in a sentence or body of text by using every other word in the text. Examining text bidirectionally increases result accuracy. This type is often used in machine learning models and speech generation applications.

What is language model in speech?

Language models rely on acoustic models to convert analog speech waves into digital and discrete phonemes that form the building blocks of words.



Large language Models

Challenges with Language Modeling?

Formal languages (like a programming language) are precisely defined. All the words and their usage is predefined in the system. Anyone who knows a specific programming language can understand what's written without any formal specification.

Machines only understand the language of numbers. For creating language models, it is necessary to convert all the words into a sequence of numbers. For the modellers, this is known as encodings.

How does Language Model Works?

Language Models determine the probability of the next word by analyzing the text in data. These models interpret the data by feeding it through algorithms.

The algorithms are responsible for creating rules for the context in natural language. The models are prepared for the prediction of words by learning the features and characteristics of a language. With this learning, the model prepares itself for understanding phrases and predicting the next words in sentences.

For training a language model, a number of probabilistic approaches are used. These approaches vary on the basis of the purpose for which a language model is created. The amount of text data to be analyzed and the math applied for analysis makes a difference in the approach followed for creating and training a language model.

For example, a language model used for predicting the next word in a search query will be absolutely different from those used in predicting the next word in a long document (such as Google Docs). The approach followed to train the model would be unique in both cases.

Types of Language Models:

There are primarily two types of language models:

1. Statistical Language Models

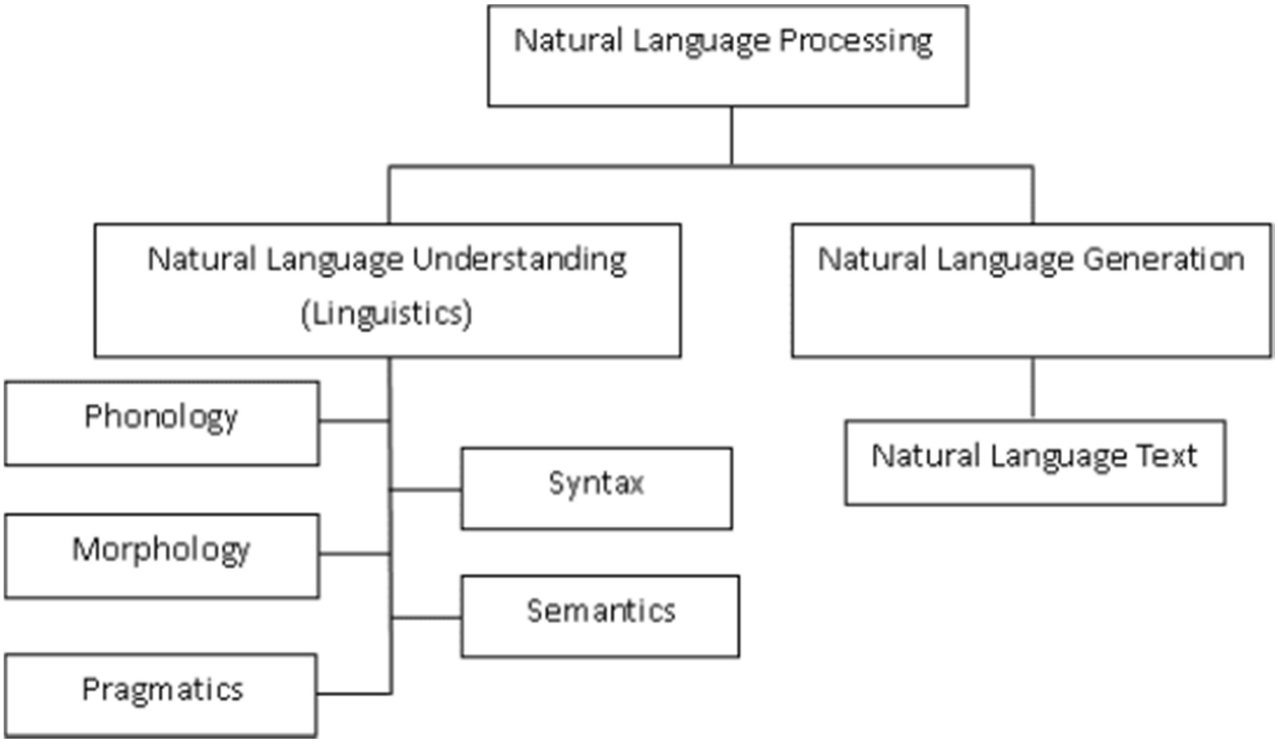
Statistical models include the development of probabilistic models that are able to predict the next word in the sequence, given the words that precede it. A number of statistical language models are in use already.

Let's take a look at some of those popular models:

- N-Gram
- Unigram
- Bidirectional
- Exponential
- Continuous Space

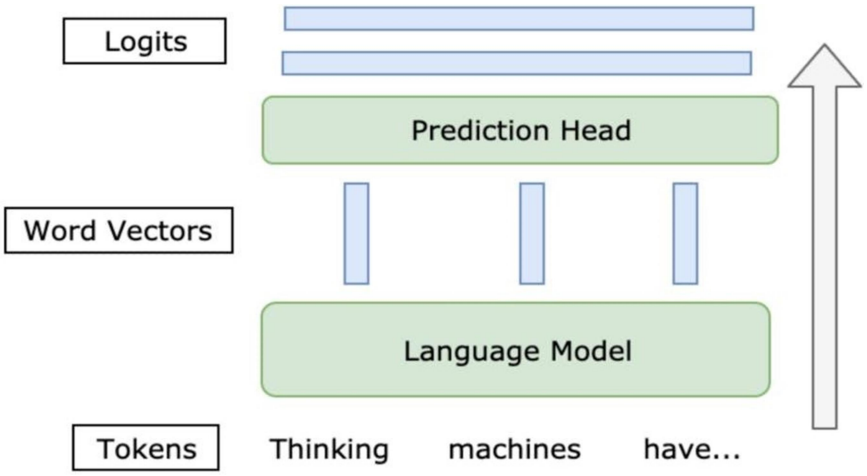
2. Neural Language Models

These language models are based on neural networks and are often considered as an advanced approach to execute NLP tasks. Neural language models overcome the shortcomings of classical models such as n-gram and are used for complex tasks such as speech recognition or machine translation.

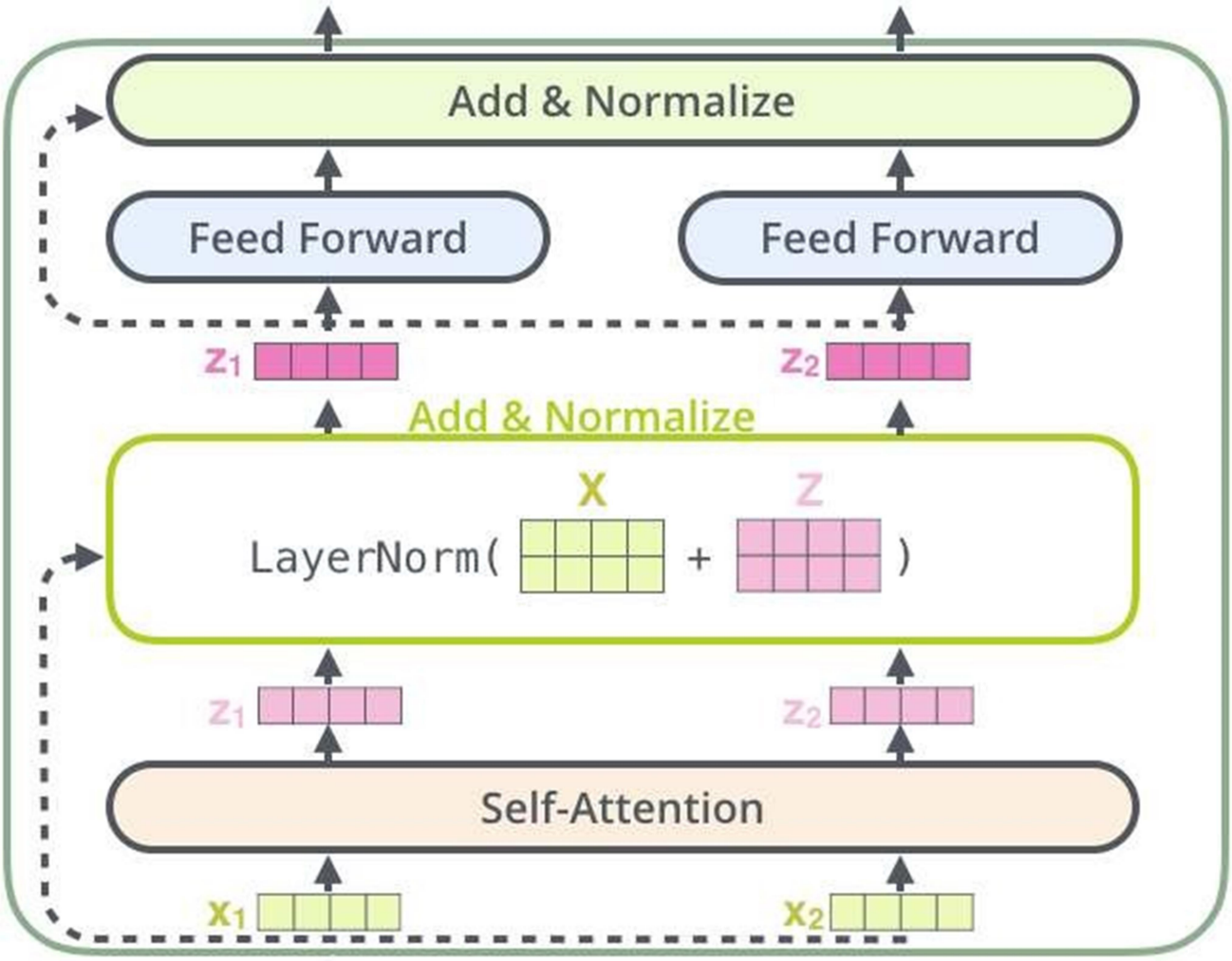


Some Common Examples of Language Models:

1. Speech Recognition
2. **Machine Translation**
3. Sentiment Analysis
4. Text Suggestions
5. Parsing Tools
6. Text Classification
7. Dialog Systems and Creative Writing
8. Text Summarization



Modern Questions Answering System



How to Train-A Question and Answering Machine Learning Models

Common Challenges in NLP Language Models:

1) Long-Term Dependency
2) Low-Resource Languages
3) Sarcasm and Irony
4) Handling Noisy Text
5) Contextual Ambiguity

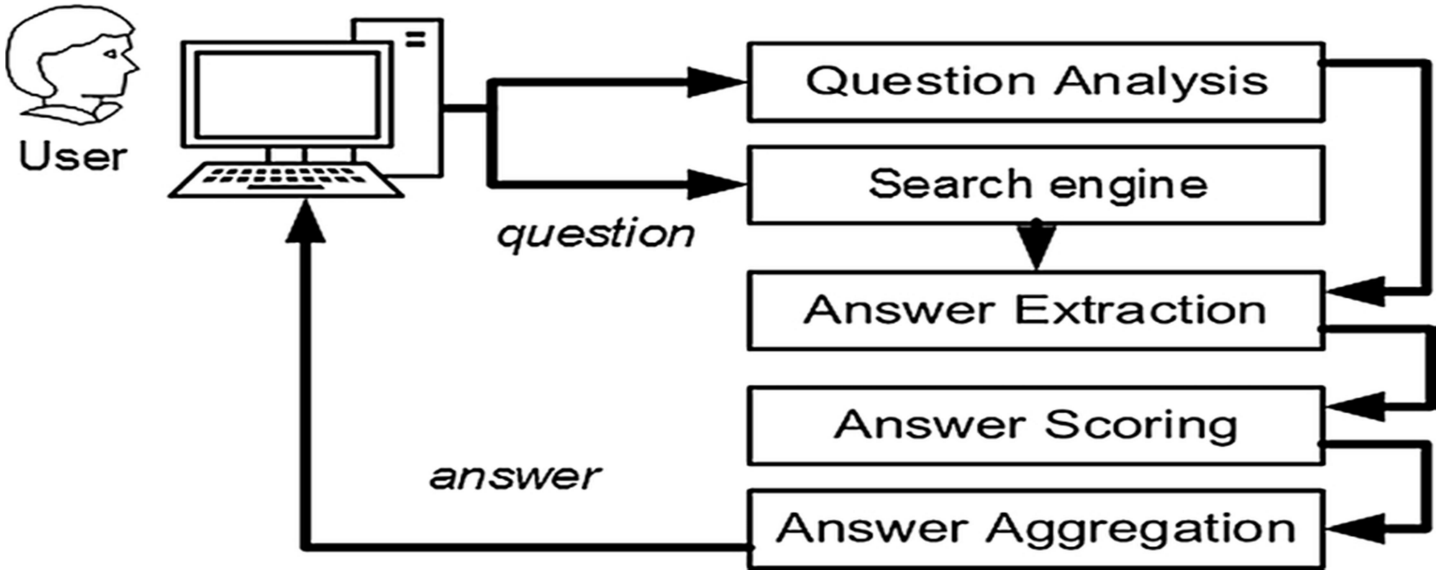
classic QA models:

Classic question-answering (QA) models in text and speech analysis have evolved over the years. Here are some of the classic models:

1. Information Retrieval (IR) Models: These models are based on retrieving relevant documents or passages from a collection in response to a query. Classic IR models include:

Vector Space Model (VSM): Represents documents and queries as vectors in a high-dimensional space and computes similarity scores between them.

Term Frequency-Inverse Document Frequency (TF-IDF): Measures the importance of a term in a document relative to a corpus.



Web based Questions and Answering Models

2.Rule-based QA Systems: These systems rely on handcrafted rules to parse questions and retrieve relevant information from structured or semi-structured data sources. Classic examples include:

ELIZA: A rule-based natural language processing program that simulates a conversation by following patterns and rules.

ALICE: Another early chatbot that uses pattern matching and predefined responses.

3.Statistical QA Models: These models utilize statistical techniques to analyze text and generate answers. Classic examples include:

IBM Watson: Utilizes a combination of statistical techniques, natural language processing, and machine learning to understand and answer questions.

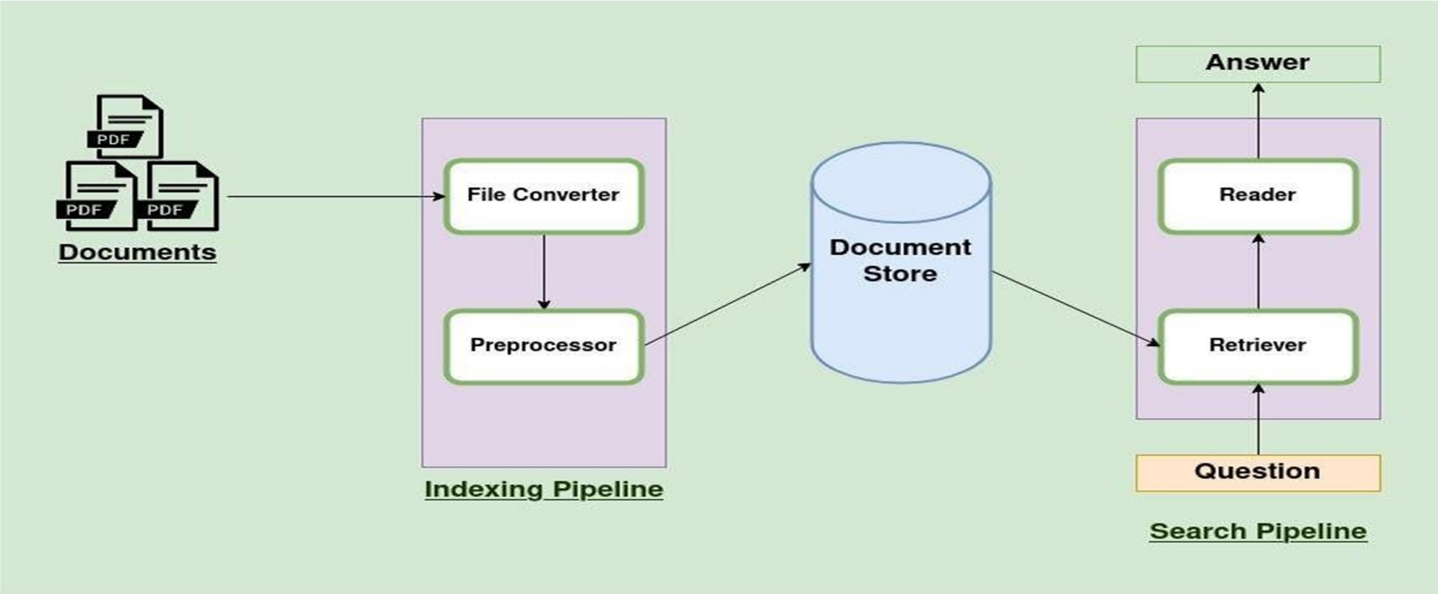
DeepQA: The architecture behind IBM Watson, which combines various algorithms and techniques for question answering.

4.Neural QA Models: These models leverage neural networks to understand and answer questions. Classic examples include:

Memory Networks: Models designed to store and retrieve information from memory, useful for tasks like question answering.

Attention Mechanisms: Mechanisms that allow neural networks to focus on relevant parts of the input, improving performance in QA tasks.

Transformer-based Models: Models like BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) have shown significant advancements in QA tasks.



Questions and Answering – NLP Projects

5. **Graph-based QA Models:** These models represent text or knowledge as graphs and perform reasoning over them to answer questions. Classic examples include:

Knowledge Graphs: Represent structured knowledge as graphs and perform graph-based reasoning to answer questions.

Graph Neural Networks (GNNs): Neural networks designed to operate on graph-structured data, which can be used for QA tasks involving graph representations.

Each of these classic models has its strengths and weaknesses, and modern QA systems often combine multiple approaches for improved performance.

What are the uses of question answering system?

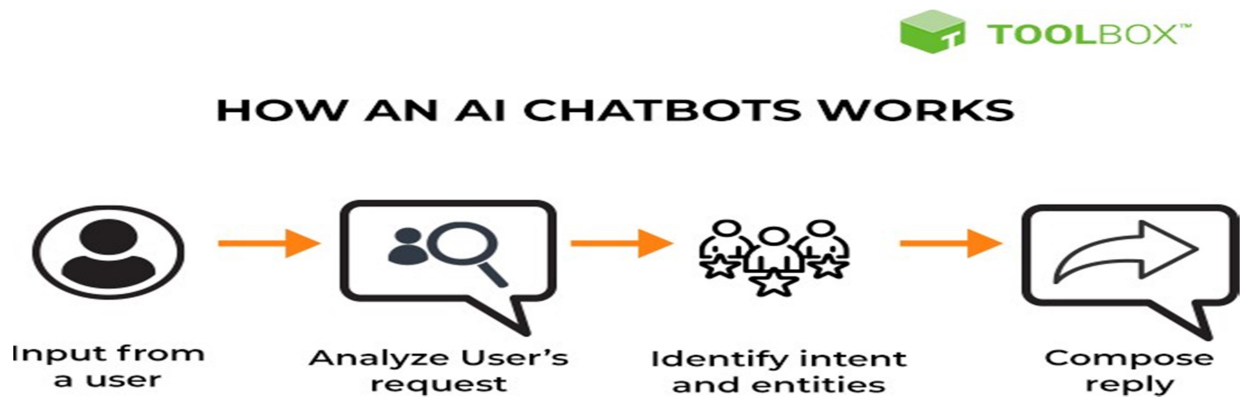
Question answering is commonly used to build conversational client applications, which include social media applications, chat bots, and speech-enabled desktop applications.

What are the 5 applications of NLP?

NLP business applications come in different forms and are so common these days. For example, spell checkers, online search, translators, voice assistants, spam filters, and autocorrect are all NLP applications.

Chatbots:

The role of chatbots in NLP lies in their ability to understand and respond to natural language input from users. This means that rather than relying on specific commands or keywords like traditional computer programs, chatbots can process human-like questions and responses.



What are the main types of chatbots?

Depending on their capabilities, chatbots can be simple, intelligent, and hybrid.

1. **Simple bots** are quite basic tools that rely on natural language processing. They can understand and respond to human queries with certain actions that are based on keywords and phrases. This type of bots has a defined rule-based decision tree (or RBDT), which helps users find needed information. FAQ chatbot is a perfect example of a simple bot.

2. **Intelligent chatbots**, which are also known as virtual assistants or virtual agents, are powered by artificial intelligence and are much more complicated than simple chatbots. They can understand human written and oral language and, which is more important, the context behind it.

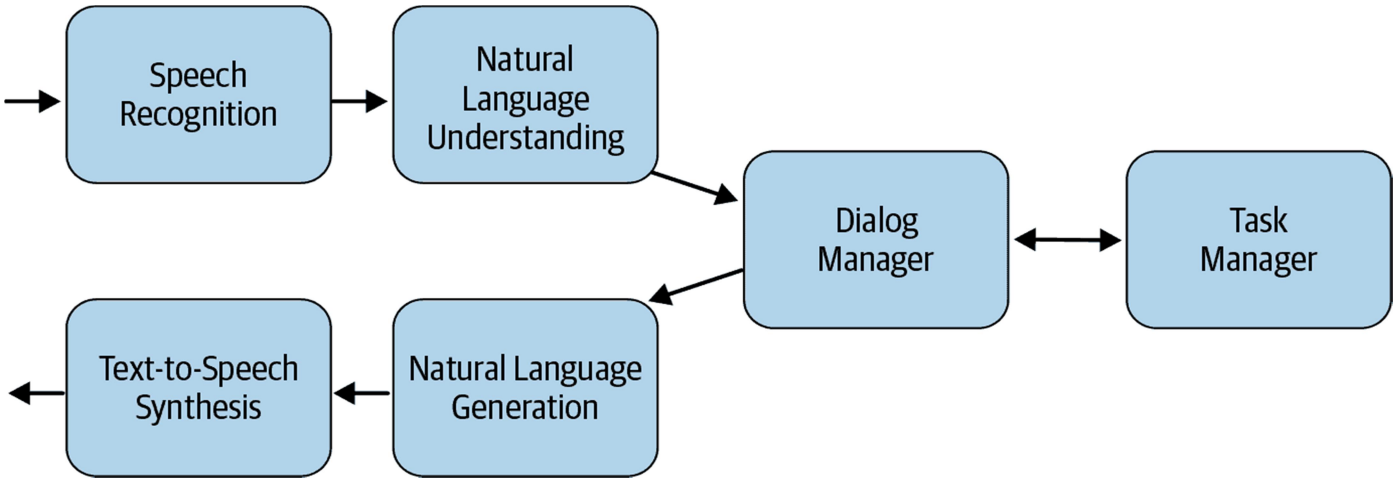
Hybrid chatbots are bots that are partially automated, meaning that they lead conversations until a human interaction is required. They might have the same functionality as simple bots, but a user can opt for a person when needed.

Chatbots can be powerful tools in text and speech analysis due to their ability to process large amounts of data quickly and efficiently. Here's how they're used:

1. **Text Analysis**: Chatbots can analyze text data to extract valuable insights such as sentiment analysis, topic modeling, keyword extraction, and named entity recognition. They can

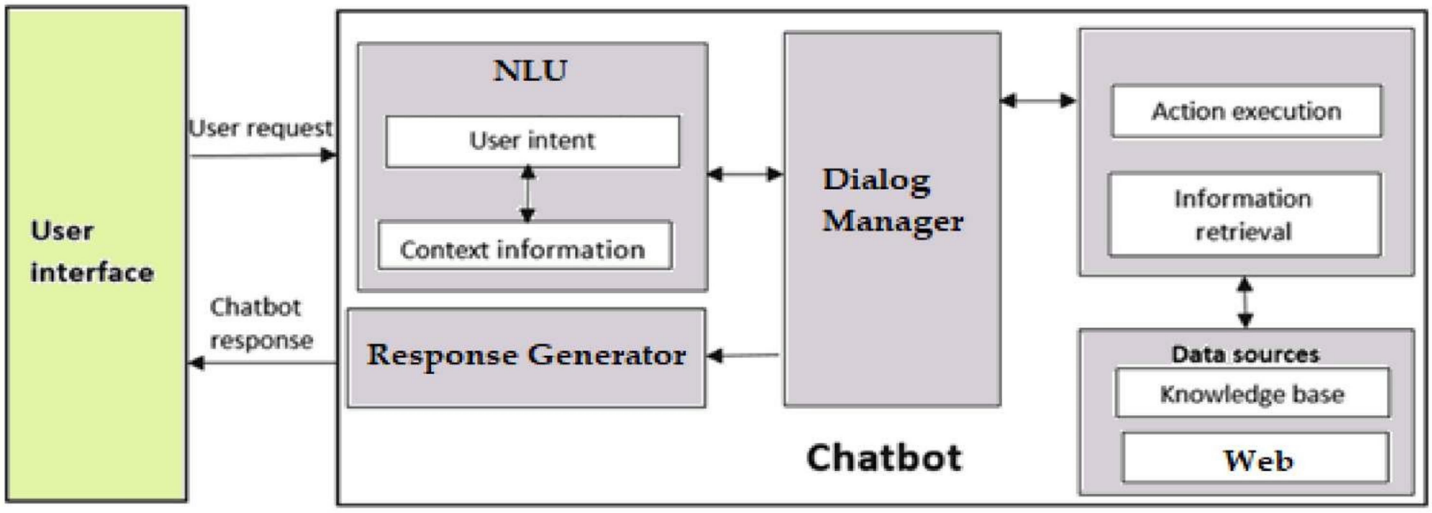
understand the context of the conversation and provide relevant responses or take appropriate actions based on the analysis.

2.Speech Recognition: With advancements in natural language processing (NLP) and speech recognition technology, chatbots can transcribe spoken language into text. This text data can then be further analyzed using text analysis techniques mentioned above.



3.Sentiment Analysis: Chatbots can analyze the sentiment expressed in text or speech, helping businesses gauge customer satisfaction, detect issues, or monitor public opinion about their products or services.

4.Customer Support: Chatbots are commonly used in customer support to analyze customer queries and provide appropriate responses. They can understand the intent behind the customer's message and either provide a solution or escalate the query to a human agent if necessary.



5. **Market Research:** Chatbots can be deployed to gather and analyze textual data from social media, forums, or surveys to understand consumer preferences, trends, and feedback on products or services.

6. **Language Translation:** Chatbots equipped with language translation capabilities can analyze and translate text or speech from one language to another, facilitating communication across linguistic barriers.

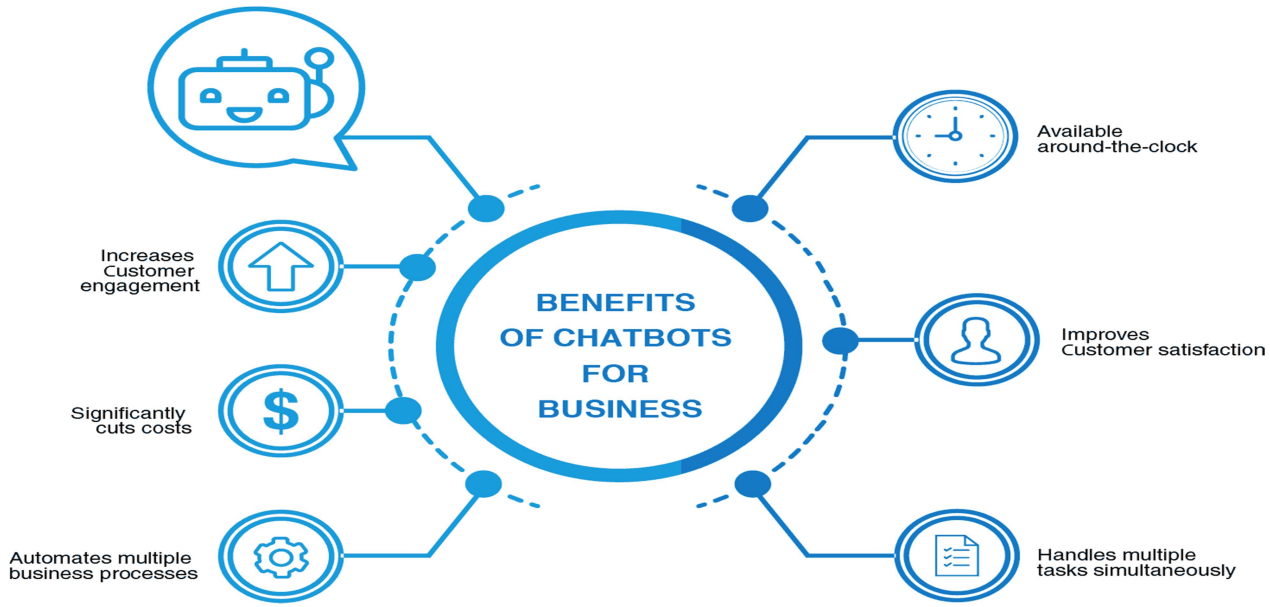
7. **Personalization:** By analyzing user interactions and preferences, chatbots can personalize responses and recommendations, improving user experience and engagement.

Chatbots Terminologies:

- Quick reply
- Hybrid Chat
- Intent
- Sentiment analysis
- Compulsory input
- Optional input
- Decision trees

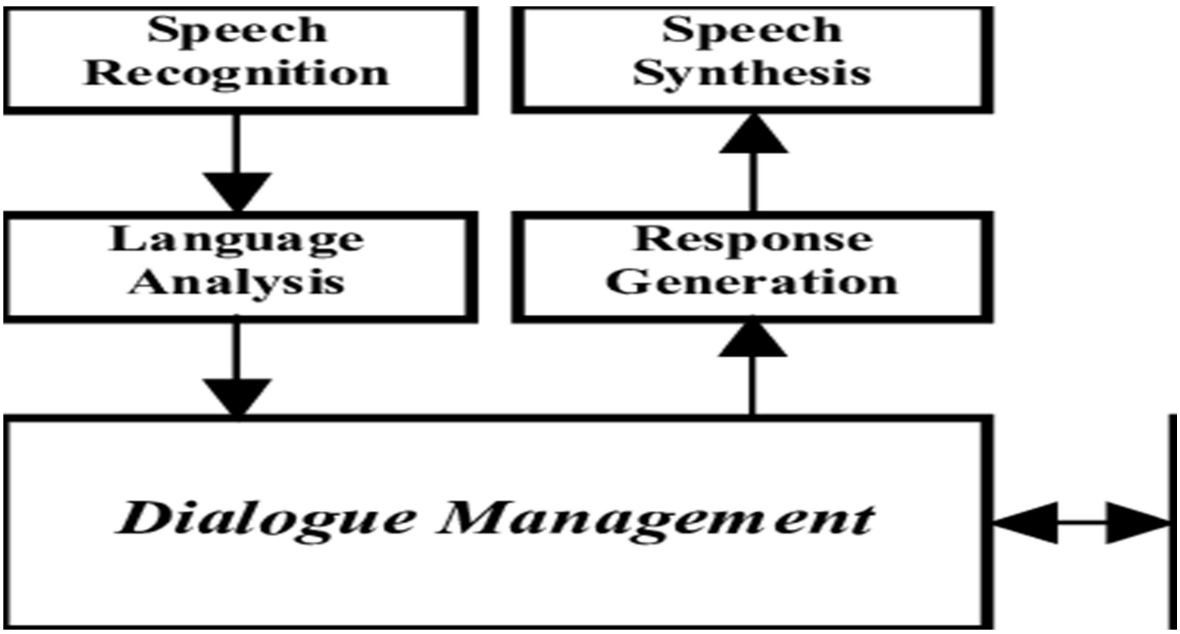
Benefits of commercial chatbots:

- Help customers find what they need much faster
- Can easily substitute a seller
- Always available at your customers' fingertips



Design of dialogue systems:

A Dialogue System is a system which interacts with human in natural language. At present many universities are developing the dialogue system in their regional language.Dialogue systems, also known as conversational agents or chatbots, are designed to interact with users in a natural and human-like manner. They can be implemented in various forms, including text-based interfaces like messaging apps or speech-based interfaces like virtual assistants.



Architecture of spoken dialogue systems

What is speech dialog system?

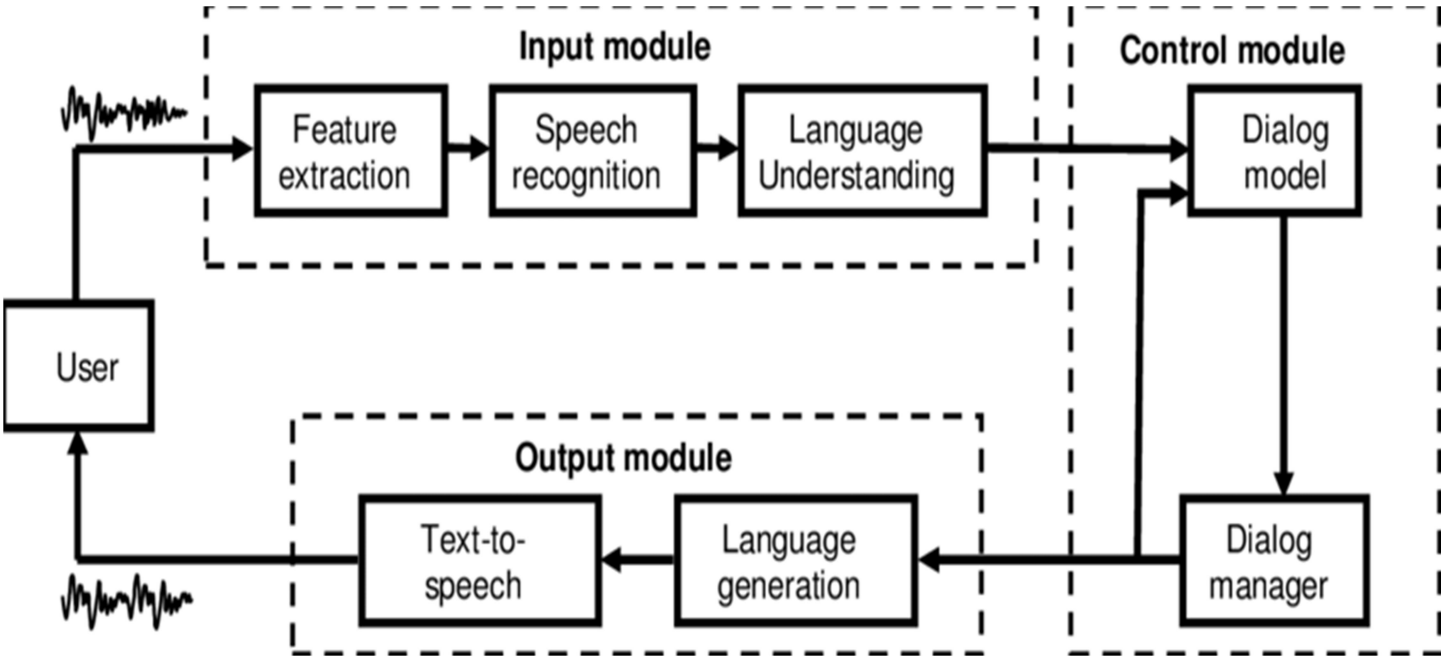
A spoken dialog system (SDS) is a computer system able to converse with a human with voice. It has two essential components that do not exist in a written text dialog system: a speech recognizer and a text-to-speech module (written text dialog systems usually use other input systems provided by an OS).

What is an example of a dialogue system?

Examples of dialogue systems in action include chatbots, food ordering apps, website AI assistants, automated customer support service, self-checkout systems, etc.

What are types of dialogue systems?

- Rule-based systems,
- Statistical systems,
- Neural networks



Components of Dialogue System:

A Dialogue system has mainly seven components. These components are following:

- Input Decoder
- Natural Language Understanding
- Dialogue Manager
- Domain Specific Component
- Response Generator
- Output Renderer

several key components:

1.Natural Language Understanding (NLU):

NLU is crucial for dialogue systems to comprehend user inputs accurately. It involves tasks such as intent classification, entity recognition, and sentiment analysis.

In text analysis, techniques like natural language processing (NLP) and machine learning models are used to parse and understand the meaning of user messages.

In speech analysis, automatic speech recognition (ASR) systems convert spoken language into text, which is then processed using NLU techniques.

2. Dialogue Management:

Dialogue management is responsible for determining the system's response based on the user's input and the current context of the conversation.

In text-based systems, dialogue management often involves maintaining a conversation state, tracking the dialogue history, and selecting appropriate responses using rule-based systems or machine learning algorithms.

In speech-based systems, dialogue management may also incorporate speech recognition results and handle interruptions or errors in speech input.

3. Response Generation:

Response generation involves creating human-like responses to user inputs. This can be achieved using templates, rule-based systems, or machine learning models like neural networks.

In text-based systems, response generation may involve generating text using language generation techniques such as neural language models (e.g., GPT).

In speech-based systems, text-to-speech (TTS) synthesis is used to convert textual responses into spoken language.

4. User Experience (UX) Design:

UX design focuses on creating a smooth and intuitive interaction between users and dialogue systems.

In text-based systems, UX design includes considerations such as message formatting, response timing, and error handling.

In speech-based systems, UX design involves designing voice prompts, handling interruptions gracefully, and providing feedback through speech.

5. Feedback and Adaptation:

Dialogue systems should be able to learn and adapt based on user feedback to improve their performance over time.

Techniques such as reinforcement learning can be used to optimize dialogue policies based on user interactions.

In text-based systems, sentiment analysis can be used to gauge user satisfaction and adjust system behavior accordingly.

In speech-based systems, user feedback can be collected through voice commands or post-interaction surveys.

6. Multi-Modality:

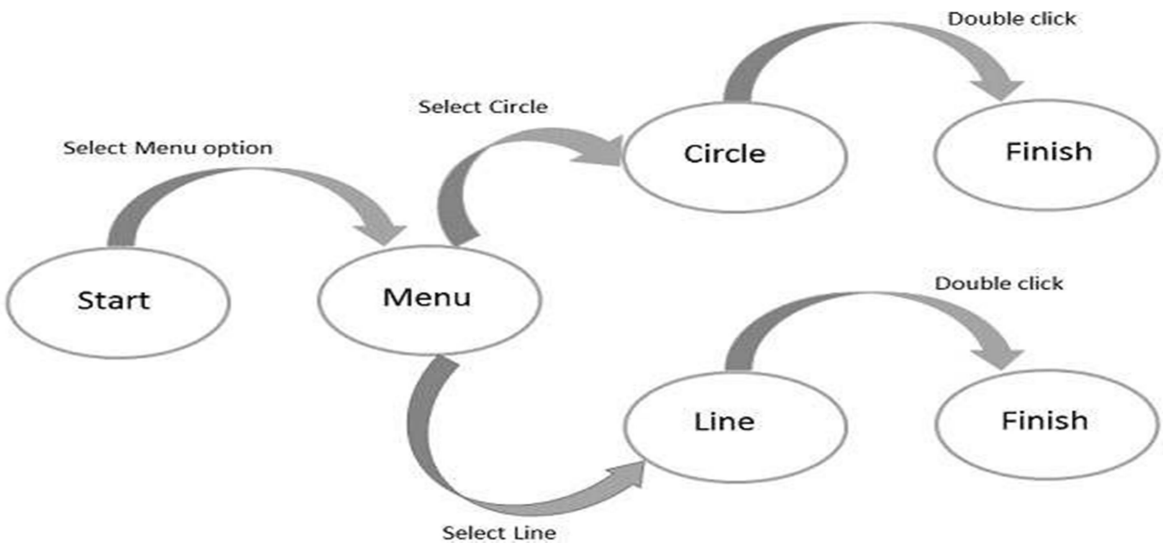
Some dialogue systems incorporate both text and speech modalities to provide a more versatile user experience.

Multi-modal systems must seamlessly integrate text and speech processing components while maintaining consistency across modalities.

7. Privacy and Security:

Dialogue systems often handle sensitive information, so ensuring privacy and security is paramount.

Techniques such as end-to-end encryption and secure data handling practices should be implemented to protect user data.



Dialog Design

Classification of Dialogue System:

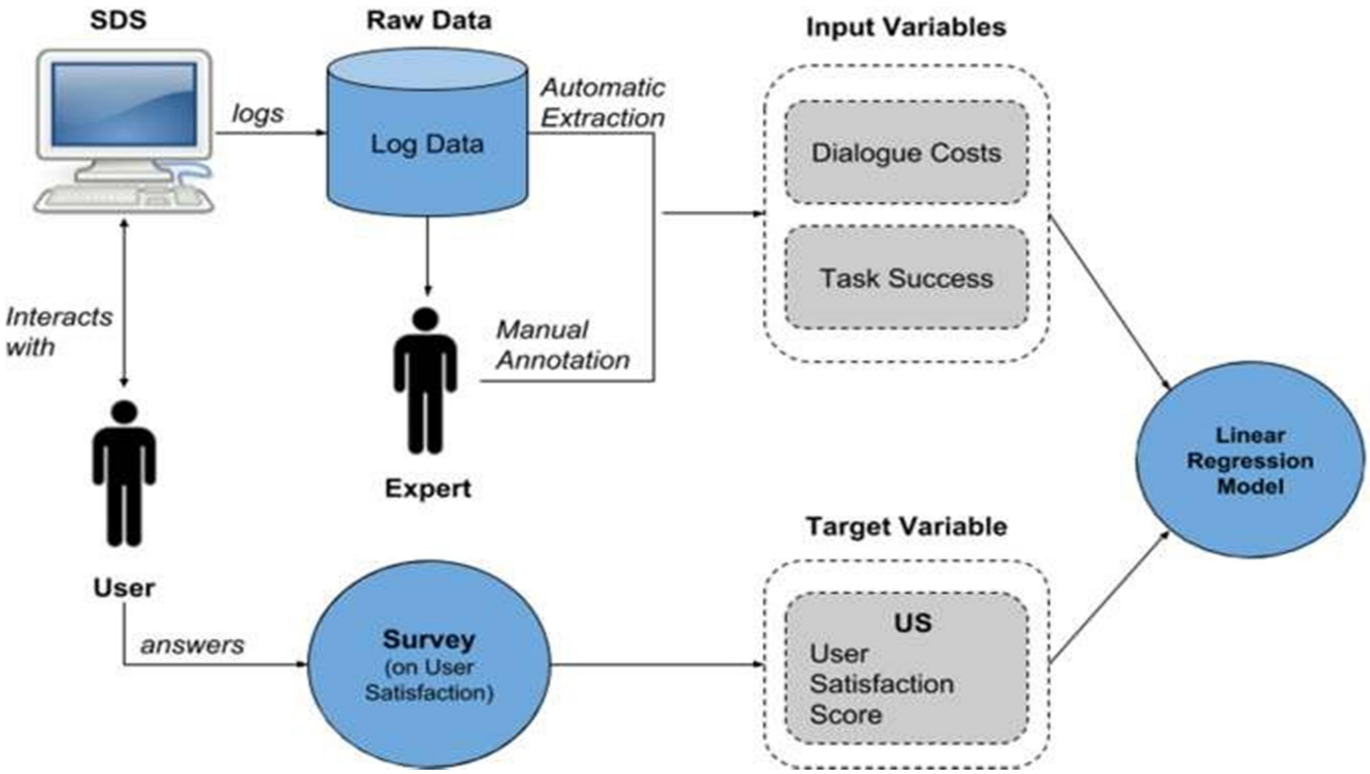
- Finite State (or graph) based systems
- Frame based systems
- Agent based systems

Evaluating dialogue systems:

What is the dialogue system architecture?

While the architecture of Dialogue Systems can vary, they typically follow the same sequence of phases: Input Recognition, Natural Language Understanding, Dialogue Management, Response Generation, and Output Rendering.

Evaluating dialogue systems, whether they operate through text or speech analysis, involves assessing various aspects of their performance, including accuracy, effectiveness, user satisfaction, and scalability. Here are some common evaluation metrics and methodologies for both text and speech-based dialogue systems:



Survey on evaluation methods for Dialog Systems

Text-Based Dialogue Systems:

Accuracy Metrics:

Intent Classification Accuracy: Measures how accurately the system classifies user intents based on their input.

Entity Recognition F1 Score: Evaluates the system's ability to correctly identify and extract entities from user messages.

Response Generation Quality: Assess the coherence, relevance, and grammatical correctness of the generated responses using metrics like BLEU, ROUGE, or human judgment.

Effectiveness Metrics:

Task Completion Rate: Determines the percentage of user queries or tasks successfully completed by the system without errors.

Response Latency: Measures the time taken by the system to respond to user inputs, aiming for low latency to improve user experience.

User Satisfaction:

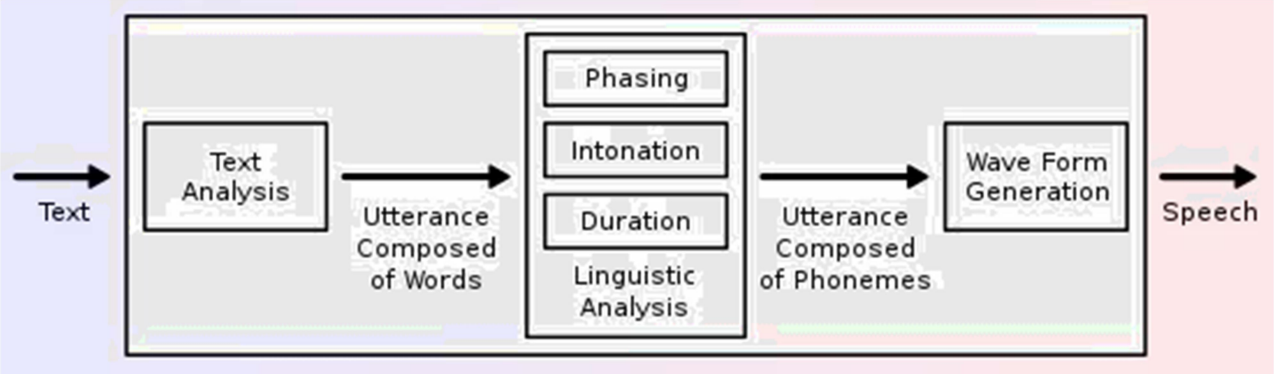
User Surveys: Collect feedback from users through surveys to assess their satisfaction with the dialogue system's performance, usability, and helpfulness.

User Ratings: Users can rate their interactions with the system on a scale, providing quantitative feedback on their satisfaction levels.

Error Analysis: Analyze common errors made by the system, such as misclassification of intents, incorrect entity recognition, or nonsensical responses, to identify areas for improvement.

Robustness and Adaptability:

Evaluate how well the system handles variations in user input, including typos, slang, or ambiguous language.



Speech Synthesis

<u>Speech-Based Dialogue Systems:</u>

Speech Recognition Accuracy:

Word Error Rate (WER): Measures the accuracy of the system's speech recognition component by comparing the transcribed text with the ground truth.

Phoneme Error Rate (PER): Evaluates the accuracy of phoneme-level transcription in speech recognition.

Naturalness of Speech Synthesis:

Mean Opinion Score (MOS): Collect subjective ratings from human listeners on the naturalness, intelligibility, and overall quality of synthesized speech.

Task Completion Rate:

Similar to text-based systems, assess the percentage of user queries or tasks successfully completed by the system through spoken interactions.

Speech Interaction Latency:

Measure the time taken by the system to process spoken input, recognize intents, generate responses, and synthesize speech, aiming for minimal latency.

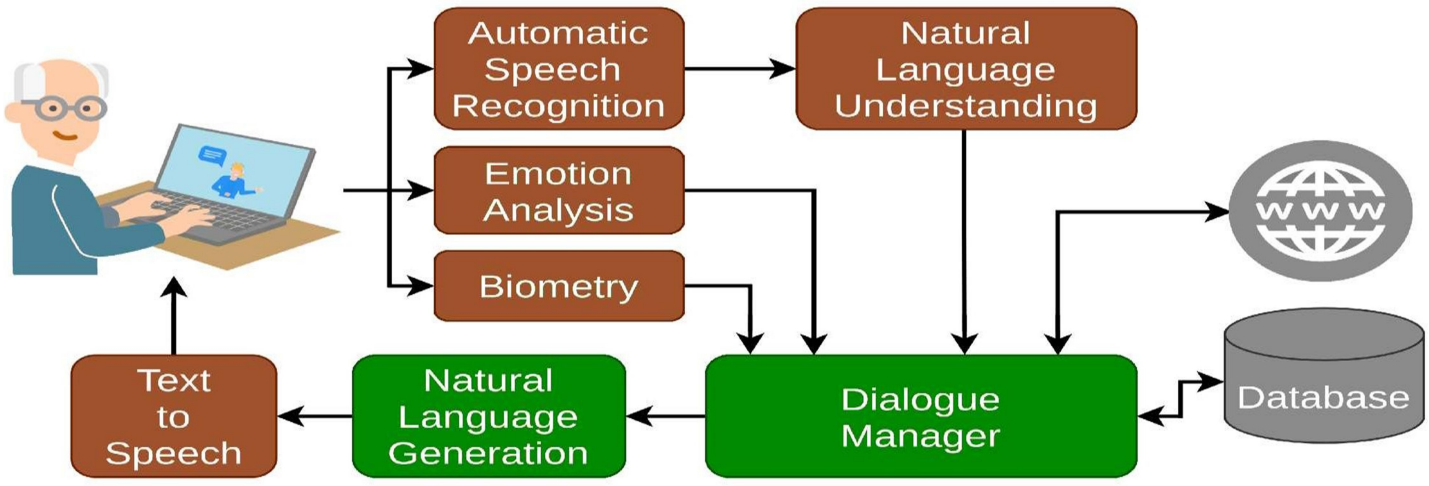
Noise Robustness:

Evaluate the system's performance in noisy environments by introducing background noise and assessing its impact on speech recognition accuracy and speech synthesis intelligibility.

User Experience in Speech-Based Interactions: Conduct user studies to gather feedback on the ease of use, naturalness, and effectiveness of speech-based interactions with the system.

Multimodal Integration:

Assess the effectiveness of integrating speech recognition and synthesis with other modalities, such as text-based input and output, to provide a seamless user experience across multiple channels.



Dialog Management and Language Generation