

QUESTION BANK

CHAPTER 1

INTRODUCTION AND WORDS ANALYSIS

PART – A

1. What is ARPAbet?

ARPAbet (or ARPAbet) is a phonetic transcription code (ASCII-based) developed by ARPA, used to represent phonemes (speech sounds) of General American English using short ASCII sequences.

2. Define wavefile format in speech processing.

The wave file (WAV) format is a standard audio file container (based on RIFF) which typically stores uncompressed PCM audio samples (e.g. linear pulse-code modulation).

3. What is a phonetic dictionary?

A phonetic dictionary (also called pronunciation lexicon) maps words (orthographic forms) to their phonetic (or phonemic) transcriptions (e.g. in ARPAbet). It is used in speech systems to know how to pronounce or recognize each word.

4. Mention any two features of the PRAAT tool.

- It can compute and display spectrograms, pitch, formants, and other acoustic measures.
- It supports scripting / batch processing to automate phonetic analysis.

5. Give an example of a phoneme in ARPAbet.

Example: “IH” is an ARPAbet phoneme (as in the vowel sound in “bit”) (Or “AH” for the vowel in “but”).

6. What is Automatic Speech Recognition?

Automatic Speech Recognition (ASR) is the process by which a machine (computer) converts spoken language (audio signal) into its corresponding textual representation (words).

7. Mention any two applications of ASR.

- Voice assistants / virtual assistants (e.g. Alexa, Siri)
- Dictation / speech-to-text for document entry

8. List basic components of an ASR system.

Some basic components are:

- Front-end feature extraction (e.g. MFCC, PLP)
- Acoustic model (mapping features to phonetic units)
- Pronunciation / lexicon (phonetic dictionary)
- Language model (predicting sequences of words)
- Decoder (search algorithm combining acoustic + language model)

9. What is the role of language models in ASR?

The language model captures probabilities of word sequences (how likely a word follows another). It helps the ASR system choose the more plausible word sequence among acoustic alternatives, improving recognition accuracy.

10. Define speaker-dependent ASR.

A speaker-dependent ASR system is one that is trained and tuned for a specific speaker's voice (i.e. requires training for that speaker), and works best (or only) for that speaker.

11. What is a Hidden Markov Model (HMM)?

An HMM is a statistical model in which the system is assumed to be a Markov process with hidden (unobserved) states; observable outputs are probabilistically generated from those hidden states.

12. Mention the three problems associated with HMM.

The classical trio of HMM problems are:

- Evaluation (computing probability of an observation sequence given model)
- Decoding (finding the best hidden state sequence for a given observation)
- Learning (estimating model parameters to maximize likelihood)

13. Define emission probability.

Emission probability is the probability of observing a particular output symbol (or observation) given a hidden state in an HMM, i.e. $b_j(o_t) = P(o_t | \text{state } j)$

14. What is a state transition in HMM?

State transition is the probability of moving from one hidden state to another in the next time step (i.e. $a_{ij} = P(\text{state}_{t+1} = j | \text{state}_t = i)$).

15. Give one application of HMM in speech.

HMMs are widely used in ASR (acoustic modeling / phoneme modeling) to model temporal sequences of speech features and infer phoneme sequences.

PART – B

1. Explain the structure and use of ARPAbet in phonetic transcription.
2. Describe wave file formats used in speech processing with an example.
3. Discuss the role and structure of phonetic dictionaries in ASR systems.
4. Demonstrate how PRAAT is used for speech analysis and annotation.
5. Explain the architecture of a typical ASR system.
6. Discuss various challenges in building ASR systems.
7. Evaluate the differences between speaker-dependent and speaker-independent ASR systems.
8. Analyze the working of ASR in noisy environments.
9. Describe HMM-based speech recognition with a suitable diagram.
10. Explain how the Viterbi algorithm is used in HMM for ASR.

CHAPTER 2

SPEECH AND SYNTACTIC

PART A

1. What is ARPA bet?

A phonetic transcription scheme using ASCII codes, developed by ARPA, to represent phonemes (speech sounds) for English.

2. Wave file format in speech processing.

WAV (wave) is an audio file format (based on RIFF) that stores uncompressed PCM audio signals, often used for storing speech waveforms in ASR systems.

3. What is a phonetic dictionary?

A lexicon that maps written words to their phonetic or phonemic transcriptions (e.g. in ARPAbet) for use by speech systems.

4. Feature of PRAAT tool (one).

It can display waveforms, spectrograms, pitch, formants and other acoustic measures.

5. Another feature of PRAAT.

It supports scripting / batch processing to automate acoustic analyses over multiple files.

6. Example of a phoneme in ARPAbet.

“IH” (as in the vowel sound in “bit”) is an ARPAbet phoneme.

7. What is Automatic Speech Recognition (ASR)?

The process of converting spoken utterances (audio signals) into their corresponding textual representation automatically.

8. Application of ASR (one).

Voice assistants (e.g. Siri, Alexa, Google Assistant).

9. Another application of ASR.

Speech-to-text transcription / dictation (for typing by voice).

10. Basic components of an ASR system.

Feature extraction, acoustic model, pronunciation lexicon (phonetic dictionary), language model, decoder / search engine.

11. Role of language models in ASR.

They assign probabilities to sequences of words, guiding the ASR system to prefer more plausible word sequences given acoustic evidence.

12. Define speaker-dependent ASR.

An ASR system that is trained specifically for one speaker’s voice and works best (or only) for that speaker.

13. What is a Hidden Markov Model (HMM)?

A statistical model where the system is modeled as a Markov process over hidden states, and each hidden state probabilistically emits observable outputs

14. Three problems associated with HMM

- Evaluation (computing probability of observations)
- Decoding (finding most likely hidden state sequence)
- Learning / parameter estimation (training model parameters)

15. Define emission probability

The probability of generating a particular observed output given that the system is in a particular hidden state.

16. What is a state transition in HMM?

The probability of moving from one hidden state to another in the next time step.

17. Application of HMM in speech.

Modeling sequences of phonemes / states in ASR to infer the underlying phonetic sequence from observed acoustic features.

18. Gaussian Mixture Model (GMM) acoustic models - definition.

A GMM is a probabilistic model that represents data as a weighted mixture of multiple Gaussian distributions, used to model emission probabilities in continuous observation spaces.

19. Embedded training (in ASR).

A training technique where alignment between speech frames and HMM states is done implicitly during training (without forced alignment), jointly optimizing segmentation and model parameters.

20. Discriminative training (in ASR).

A method of training models (e.g. acoustic models) by optimizing a discriminative criterion (e.g. Minimum Classification Error, Maximum Mutual Information) rather than just maximizing likelihood, thereby improving classification accuracy in ASR.

PART – B

1. Explain the working of HMM-based speech recognition and discuss the role of Gaussian Mixture Models in acoustic modeling.
2. Describe the concept of Context-Free Grammars (CFG) and explain how Treebanks are used in syntactic parsing.
3. Compare and contrast Dependency Grammar with Phrase Structure Grammar. How is syntactic parsing performed using these frameworks?
4. What is discriminative training in speech recognition? Discuss its advantages over generative training methods.
5. Explain the importance of phonetic dictionaries and wave file formats in automatic speech recognition systems.
6. How do tasks like stemming, stop-word removal, indexing, and ranking work?
7. Describe in detail the process of syntax-driven semantic analysis.

8. Outline the architecture of a syntax-driven analyzer that produces logical forms from natural language input.
9. Explain what is meant by the internal structure of lexical items and phrases in semantics.
10. Given a natural language query and a large corpus of documents, describe how information retrieval can benefit from deep semantic analysis (including FOPC representations, WSD, sense matching)
11. Compare and contrast human speech recognition and automatic speech recognition.
12. Describe how CFGs are extracted or probabilistically weighted from treebanks. What are limitations of CFGs in representing all natural language phenomena?
13. Design an end-to-end pipeline for a speech-to-text system that includes acoustic modeling, language modeling, lexical resources, and syntactic parsing.
14. Explain each component, how they interact, what challenges arise at each step, and how you would mitigate errors (e.g. using robust analysis, error correction, or re ranking).
15. Conversely, what are the advantages of ASR? What are the main challenges in bridging the gap between human and machine speech recognition?

CHAPTER 3

SYNTAX

PART – A

1. What is First Order Predicate Calculus?

It is a logic system that extends propositional logic by allowing quantification over individual variables (using \forall and \exists) and predicates relating those individuals.

2. What is a “predicate” in predicate calculus?

A predicate is a symbol (or relation) that denotes a property or relation among individuals, e.g. $P(x)$, $R(x,y)$.

3. What does “representing linguistically relevant concepts” refer to?

Encoding entities, properties, events, relations, and temporal or modal features of natural language in a formal representation (e.g. predicates, functions).

4. What is syntax-driven semantic analysis?

A method where semantic interpretation is guided by the syntactic parse structure; meaning is composed bottom-up following the syntax tree.

5. What are semantic attachments?

Logical form fragments or meanings associated with syntactic nodes or lexical items (e.g. λ -expressions) used in composing sentence meaning.

6. What is a syntax-driven analyzer?

A system whose semantic module follows the syntactic parser: when a parse rule is applied, the corresponding semantic composition rule is applied.

7. What is robust analysis in semantics?

Techniques to produce meaningful interpretations even when input is noisy, ungrammatical, ambiguous, or incomplete.

8. Define lexeme.

A lexeme is the base or canonical form of a word (dictionary headword) abstracting away from inflections.

9. What is a sense (in lexical semantics)?

One distinct meaning or interpretation of a lexeme (e.g. “bank” as financial institution vs “bank” as river edge).

10. What is internal structure (in lexical semantics)?

The semantic decomposition of a word or phrase into more primitive semantic components or roles.

11. Define word sense disambiguation (WSD).

The process of selecting the correct sense of an ambiguous word within a specific context.

12. What is information retrieval (IR)?

The task of retrieving relevant documents or information in response to a user query from a large repository.

13. Give an example of a predicate logic representation of a simple sentence.

E.g. “All dogs bark” $\rightarrow \forall x (\text{Dog}(x) \rightarrow \text{Bark}(x))$.

14. What is a challenge when representing modals or tense in FOPC?

FOPC has no built-in modal or temporal operators, making representation of “might”, “will”, “before”, etc. difficult.

15. Why is syntax-driven semantics useful in NLP?

It ensures alignment between syntactic structure and meaning composition, leading to systematic, modular semantic derivation.

16. What is a problem for purely syntax-driven systems?

Non-compositional idioms, long-distance dependencies, or context effects that the syntax alone cannot resolve.

17. Why do we need WSD for information retrieval?

To disambiguate query and document terms so matching is more semantically precise (reducing irrelevant matches)

18. What is underspecified semantics (in robust analysis)?

Representing ambiguous or unresolved semantic structure temporarily (postponing disambiguation) to avoid failure.

19. How might lexemes and senses be stored in a lexical resource?

Lexicon entries list a lexeme along with its possible senses, each associated with a meaning representation or sense identifier.

20. What is sense clustering in WSD?

Grouping similar senses or meanings into clusters (reducing granularity) to make disambiguation easier or more robust.

PART - B

1. Explain First Order Predicate Calculus and its role in representing linguistically relevant concepts.
2. Discuss Syntax-Driven Semantic Analysis and the process of Semantic Attachment. How does this approach help in natural language understanding?
3. What is Word Sense Disambiguation? Describe techniques used for WSD and its importance in NLP applications.
4. Explain the concept of Lexemes and their Internal Structure in computational linguistics.
5. Describe the role of Information Retrieval in NLP and how syntax and semantics contribute to improving retrieval results.
6. Given current advances in neural networks and large language models, how can symbolic semantics (FOPC, syntax-driven semantics, WSD) be integrated with neural architectures for deeper language understanding?

7. What is semantic parsing? Describe how a system can map input sentences to domain logical forms using a syntax + semantics approach.
8. How do thematic roles, argument structure, and event decomposition factor into internal structure?
9. Describe methods like resolution, theorem proving, model building. Also discuss practical limitations (computational complexity, incomplete knowledge) and strategies (approximate inference, constraint solving).
10. Explain how you would integrate FOPC (or another formalism), WSD, semantic attachments, robust analysis, and IR components.
11. Discuss sense clustering, coarse vs fine senses, and adaptation of sense inventories to specific tasks (e.g. IR vs deep understanding). How do you handle low-resource languages or limited annotation?
12. FOPC has limitations. Describe alternative or extended semantic frameworks (e.g. Discourse Representation Theory, Minimal Recursion Semantics, graph-based meaning representations).
13. Compare their strengths and trade-offs. Give examples showing how they improve expressiveness over basic predicate logic.
14. Given current advances in neural networks and large language models, how can symbolic semantics (FOPC, syntax-driven semantics, WSD) be integrated with neural architectures for deeper language understanding?
15. Propose an architecture or research direction combining symbolic and neural semantics. What are expected benefits, challenges, and open research questions?

CHAPTER 4

SEMANTICS AND PRAGMATICS

PART – A

1. What is First-Order Logic (FOL)?

First-Order Logic (FOL) is a formal system used in mathematics, philosophy, linguistics, and computer science. It provides a framework for defining and reasoning about objects and their relationships using quantifiers (\forall , \exists), predicates, and functions.

2. What is the syntax of First-Order Logic?

The syntax of FOL includes variables, constants, predicates, functions, logical connectives (\neg , \wedge , \vee , \rightarrow , \leftrightarrow), and quantifiers. Formulas are constructed using these elements to express statements about objects and their relationships.

3. What is the semantics of First-Order Logic?

The semantics of FOL assigns meanings to the symbols and formulas. An interpretation consists of a domain of discourse and an interpretation function that maps constants, functions, and predicates to specific objects, functions, and relations within the domain.

4. What are Description Logics (DL)?

Description Logics are a family of formal knowledge representation languages used to describe and reason about the concepts and relationships in a domain. They provide a balance between expressiveness and computational tractability, making them suitable for applications like ontologies.

5. What is the syntax of Description Logics?

The syntax of DL includes concept names, role names, and individuals. Concepts can be combined using constructors like conjunction (\sqcap), disjunction (\sqcup), negation (\neg), and existential quantification (\exists). Roles represent relationships between individuals.

6. What is the semantics of Description Logics?

The semantics of DL is based on interpretations that consist of a domain of individuals and an interpretation function that maps concept names to sets of individuals and role names to sets of pairs of individuals. This allows for reasoning about subsumption, satisfiability, and instance checking.

7. What is Syntax-Driven Semantic Analysis?

Syntax-Driven Semantic Analysis is a method in computational linguistics where the syntactic structure of a sentence is used to guide the construction of its semantic representation. This approach ensures that the meaning is derived in a way consistent with the sentence's grammatical structure.

8. What are Semantic Attachments?

Semantic Attachments are annotations added to the syntactic structure of a sentence that represent its meaning. These attachments can take the form of lambda calculus expressions

or other formal representations that capture the semantic content associated with each part of the sentence.

9. What is Computational Semantics?

Computational Semantics is the field of study that focuses on the automated process of constructing and reasoning with meaning representations of natural language expressions. It bridges linguistics and computer science to enable machines to understand and process human language.

10. What is Lexical Semantics?

Lexical Semantics is the subfield of semantics concerned with the meaning of words and their relationships to each other. It involves studying word meanings, sense distinctions, and how words combine to form phrases and sentences.

11. What is Pragmatics in Linguistics?

Pragmatics is the branch of linguistics that studies how context influences the interpretation of meaning. It examines aspects like speaker intentions, presuppositions, implicatures, and how language is used in communication beyond the literal meaning.

12. What is Discourse Analysis?

Discourse Analysis is the study of larger units of language, such as conversations, written texts, or speeches. It focuses on how sequences of sentences or utterances relate to each other and how meaning is constructed across these larger units.

13. What are Dialogue Systems?

Dialogue Systems are computer systems designed to converse with humans using natural language. They can be task-oriented, aiming to complete specific tasks, or open-domain, engaging in general conversation.

14. What are Conversational Agents?

Conversational Agents, also known as chatbots or virtual assistants, are AI systems that interact with users through natural language. They can perform tasks, answer questions, and simulate conversation, often using machine learning and natural language processing techniques.

15. What is Natural Language Generation (NLG)?

Natural Language Generation is a subfield of artificial intelligence that focuses on producing meaningful and coherent text from structured data. It involves converting data representations into human-readable language, useful in applications like report generation and automated content creation.

16. What is Statistical Alignment in Machine Translation?

Statistical Alignment in Machine Translation refers to the process of aligning words or phrases in a source language with their corresponding translations in a target language. This alignment is typically learned from large bilingual corpora and is crucial for statistical machine translation models.

17. What is Word Alignment in Machine Translation?

Word Alignment in Machine Translation is the task of mapping individual words in a source language sentence to their corresponding words in a target language sentence. Accurate word alignment is essential for training effective translation models.

18. What is Text Alignment in Machine Translation?

Text Alignment in Machine Translation involves aligning entire sentences or paragraphs in a source language with their translations in a target language. This alignment helps in training machine translation systems by providing parallel corpora.

19. What is Statistical Machine Translation (SMT)?

Statistical Machine Translation is a method of machine translation that relies on statistical models to generate translations. These models are trained on large bilingual corpora and use algorithms to predict the most likely translation for a given input.

20. What are the Challenges in Statistical Machine Translation?

Challenges in Statistical Machine Translation include handling ambiguous words, maintaining grammaticality in the target language, dealing with out-of-vocabulary words, and ensuring the translated text preserves the meaning and context of the original.

PART - B

1. Explain First-Order Logic (FOL) and its application in computational linguistics.
2. Discuss Description Logics (DL) and their role in knowledge representation.
3. Elucidate Syntax-Driven Semantic Analysis and Semantic Attachments.

Natural Language Processing

4. Define Computational Semantics and its significance in NLP.
5. Analyze Lexical Semantics and its impact on word meaning.
6. Explain Pragmatics and its role in discourse and dialogue.
7. Discuss Natural Language Generation (NLG) and its methodologies.
8. Elucidate Statistical Machine Translation (SMT) and its components.
9. Compare Statistical Machine Translation with Neural Machine Translation (NMT).
10. Discuss the integration of Syntax, Semantics, and Pragmatics in NLP systems.
11. Explain Description Logics (DL) and their role in knowledge representation.
12. Explain Pragmatics and its role in discourse and dialogue.
13. Discuss Statistical Machine Translation (SMT) and its components.
14. Compare Statistical Machine Translation with Neural Machine Translation (NMT).
15. Explain the challenges in integrating Syntax, Semantics, and Pragmatics in NLP systems?

CHAPTER 5 APPLICATION PART - A

1. What is supervised learning in QA?

Supervised learning in QA involves training models on labeled datasets where each question is paired with a correct answer, enabling the model to learn patterns and relationships.

2. Define extractive QA.

Extractive QA models select a span of text directly from the context as the answer to a given question.

3. What is generative QA?

Generative QA models generate an answer in natural language, which may not be directly present in the context.

4. Name a popular dataset for training QA models.

SQuAD (Stanford Question Answering Dataset) is widely used for training and evaluating QA models.

5. What role does BERT play in QA?

BERT (Bidirectional Encoder Representations from Transformers) is pre-trained on large text corpora and fine-tuned for QA tasks to understand context and semantics.

6. Explain the concept of 'context' in QA.

In QA, 'context' refers to the passage or document from which the model extracts or generates.

7. What is the significance of 'attention mechanism' in QA models?

Attention mechanisms allow models to focus on relevant parts of the context, improving accuracy.

8. Differentiate between closed-domain and open-domain QA.

Closed-domain QA is limited to a specific topic or dataset, while open-domain QA can answer questions on any topic.

9. What is a 'passage ranking' in QA systems?

Passage ranking involves selecting the most relevant passage from a large corpus to answer a question.

10. Define 'extractive span prediction'.

Extractive span prediction involves identifying the start and end positions of the answer span within the context.

11. What is supervised learning in QA?

Supervised learning in QA involves training models on labeled datasets where each question is paired with a correct answer, enabling the model to learn patterns and relationships.

12. Define extractive QA.

Extractive QA models select a span of text directly from the context as the answer to a given question.

13. What is generative QA?

Generative QA models generate an answer in natural language, which may not be directly present in the context.

14. Name a popular dataset for training QA models.

SQuAD (Stanford Question Answering Dataset) is widely used for training and evaluating QA models.

15. What role does BERT play in QA?

Natural Language Processing

BERT (Bidirectional Encoder Representations from Transformers) is pre-trained on large text corpora and fine-tuned for QA tasks to understand context and semantics.

16. Explain the concept of 'context' in QA.

In QA, 'context' refers to the passage or document from which the model extracts or generates.

17. What is the significance of 'attention mechanism' in QA models?

Attention mechanisms allow models to focus on relevant parts of the context, improving accuracy.

18. Differentiate between closed-domain and open-domain QA.

Closed-domain QA is limited to a specific topic or dataset, while open-domain QA can answer questions on any topic.

19. What is a 'passage ranking' in QA systems?

Passage ranking involves selecting the most relevant passage from a large corpus to answer a question.

20. Define 'extractive span prediction'.

Extractive span prediction involves identifying the start and end positions of the answer within the context.

PART – B

1. Explain the concept of supervised learning in machine learning. discuss its application in natural language processing (nlp) tasks such as question answering, summarization, and machine translation.
2. Discuss the architecture and working of transformer-based models like bert and gpt in supervised learning tasks. how do they contribute to advancements in nlp?
3. What are the challenges associated with training supervised learning models for question answering tasks? discuss methods to overcome these challenges.
4. Compare and contrast extractive and abstractive summarization techniques in supervised learning. provide examples of models used for each approach.
5. Explain the process of training a supervised machine translation model. discuss the role of parallel corpora and evaluation metrics.

6. Discuss the concept of fine-tuning in supervised learning. how is it applied to nlp tasks like question answering and summarization?
7. Evaluate the impact of large-scale pre-trained models on supervised learning in nlp. discuss their advantages and limitations.
8. What is the role of evaluation metrics in supervised learning for nlp tasks? discuss commonly used metrics for question answering and summarization.
9. Analyze the ethical considerations in deploying supervised learning models for nlp tasks. discuss issues related to bias, fairness, and transparency.
10. Discuss the future directions of supervised learning in nlp. what advancements can be anticipated in question answering, summarization, and machine translation?