# Dialog Systems Exam Sample Questions

July 10, 2025

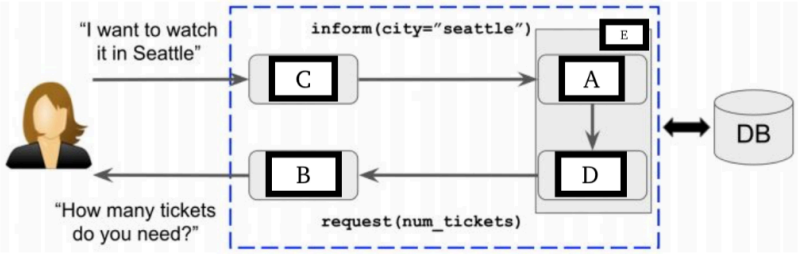



Figure 1: Task-oriented Dialog System

1. Examine the task-oriented dialog system in Figure 1. Assume the system uses text and not speech.

   (a) Write the letter of the corresponding component next to each name:

   - (POL) Dialog Policy: D
   - (NLU) Natural Language Understanding: C
   - (DST) Dialog State Tracking: A
   - (NLG) Natural Language Generation: B
   - (DM) Dialog Manager: E

   (b) Write the initials (e.g., DM for Dialog Manager) of the dialog component next to the statement that best matches it:

   - Keeps track of the users goals: DST
   - Historically involved sentence planning and surface realization: POL
   - Proposes an assignment of values to slots from an ontology: NLU
   - Typically learned through reinforcement learning and decides what action to take next: NLG

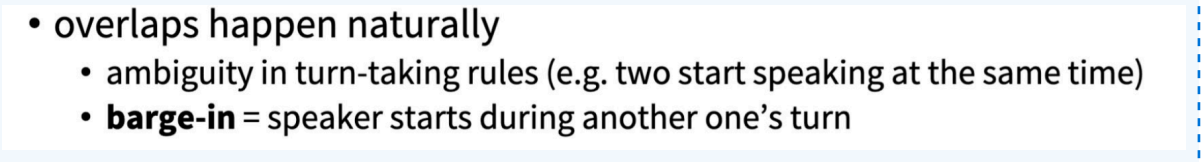
2. Name and provide examples of five types of dialog system applications.

[ Needs to be verified ]

1. **Task-oriented**:
   - Example: Booking a flight, ordering food, scheduling appointments.
2. **Open-domain conversation**:
   - Example: Casual conversation, customer support, mental health chatbots.
3. **Information-seeking**:
   - Example: Answering questions about weather, news, or general knowledge.
4. **Personal assistant**:
   - Example: Setting reminders, managing calendars, controlling smart home devices.
5. **Educational**:
   - Example: Tutoring systems, language learning assistants, interactive quizzes.

3. What is turn-taking in dialog and why is it difficult for a dialog system to handle it? Name and explain at least two reasons.

- **Turn-taking** is the process of alternating speaking turns between speakers in a conversation.




- Difficulties for dialog systems:
  1. **Complex human cues**: Systems may struggle to predict when to respond, leading to interruptions or delays.
  2. **Overlaps**: Natural conversations often have overlapping speech, which can confuse systems that expect clear turn boundaries.

4. How is speech different from text for dialog systems? Name five differences.

Natural speech is very different from written text:
1. **ungrammatical**
2. **restarts, hesitations, corrections**
3. **overlaps**
4. **pitch, stress**
5. **accents, dialect**

5. Name and explain five types of speech acts.

1. **Assertive**: Speaker commits to the truth of a proposition.
   - Example: "It's raining outside."
2. **Directive**: Speaker wants the listener to do something.
   - Example: "Stop it!"
3. **Commissive**: Speaker commits to doing something themselves.
   - Example: "I'll come by later."
4. **Expressive**: Speaker expresses their psychological state.
   - Example: "Thank you!"
5. **Declarative**: Speaker performs actions using performative verbs.
   - Example: "You're fired!"

6. Given the following exchange between A and B, which of Grice's maxims is most clearly broken? What is the implicature of this broken maxim?

   (a) Person A: Did you study enough for the exam?

   (b) Person B: I was so busy yesterday. I spent almost the entire day cleaning my house!

Circle the maxim and describe the implicature below:

Quantity, Quality, Relation, Manner

- **Broken maxims**: Quantity, Relation
- **Implicature**: Person B states too much information and does not answer the question directly, suggesting they did not study enough for the exam.

7. Describe **grounding** in dialog systems and name at least two grounding signals, each briefly explained in a sentence.

1. **Grounding** ensures mutual understanding.
2. **Grounding signals**:
   - **Implicit**: confirm without state it
   - **Explicit**: confirm message

8. What are anaphora and cataphora, and why are they problematic for dialog systems?

- **Anaphora**: refers to entities mentioned earlier in the conversation.
- **Cataphora**: refers to entities mentioned later in the conversation.
- **Problematic**: Dialog systems may struggle to resolve these references without sufficient context, leading to misunderstandings or incorrect responses.

9. What is adaptation/entrainment in dialog, and in what forms does it take place?

- **Adaptation/Entrainment** refers to the phenomenon where speakers adjust their language, style, or behavior to align with their conversational partner. It can take place in various forms, including:
  - **Lexical entrainment**: using similar words or phrases.
  - **Syntactic entrainment**: adopting similar sentence structures.
  - **Prosodic entrainment**: matching speech rate, tone, and rhythm.

10. You have developed a function to measure lexical entrainment over the turns in a conversation. You apply this to a dataset of human–machine conversations with your system and find that over time, the entrainment increases. What does this tell you about your system?

An increase in lexical entrainment over time in human-machine conversations suggests that your system is successfully adapting its word choices to align with those of the human user. This indicates that the system is becoming:

- **More efficient and natural** in the dialogue flow.
- Likely **enhancing mutual understanding** between the user and the system.
- Potentially leading to **improved user experience** and preference, as entrainment is a sign of successful interaction and can make the system's responses feel more human-like and cooperative.

11. What is the difference between a chatbot and a dialog system?

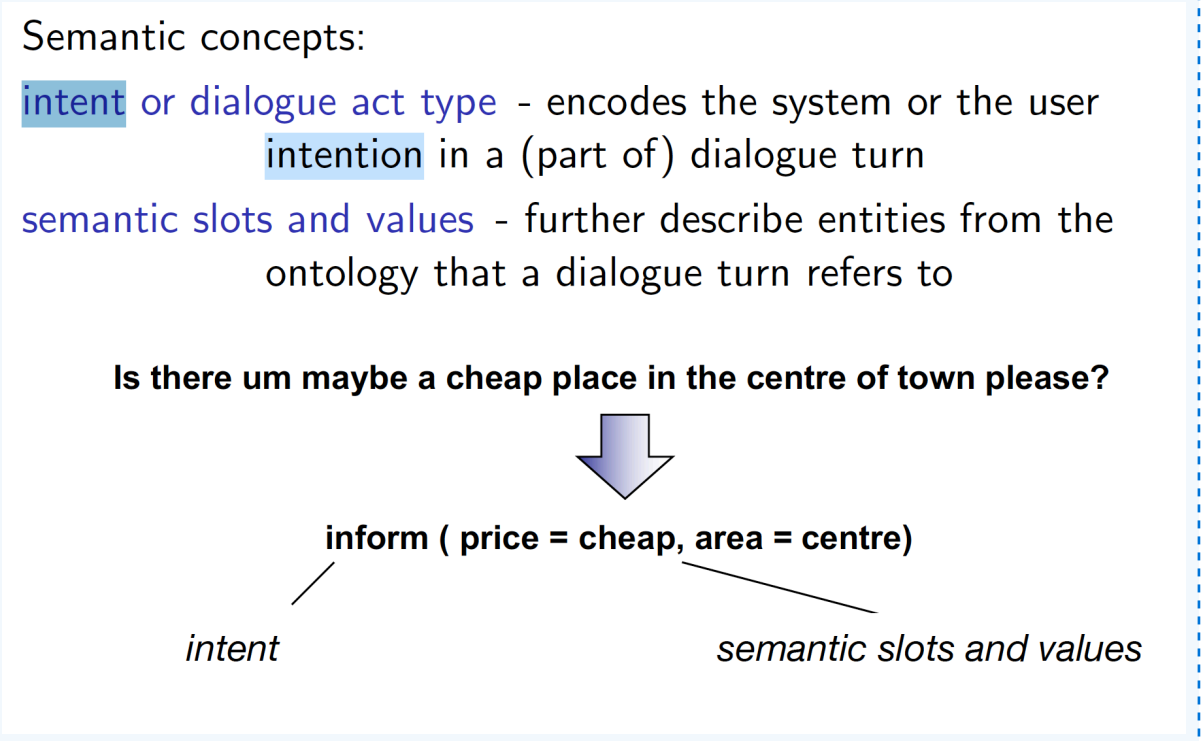
- A **chatbot** is a system that only response to the inputs relied on static knowledge.
- A **dialog system** is an interface to an external knowledge source with the main goal is to retrieve information from the knowledge source.

12. What is an ontology, and how are the ontology slots distinguished from one another?

- An **ontology** is a structured description of the external knowledge that defines the Domain, Slot, and Value.
- **Ontology slots** are distinguished by 3 types based on their values:
  - Categorical: one of the possible values
  - Binary: True/False values
  - Non-Categorical: number of values is not limited

13. Explain and give examples for the two semantic concepts in dialog acts.

1. **intent or dialogue act type**:
   - encodes the system or the user intention in a (part of) dialogue turn
2. **semantic slots and values**:
   - further describe entities from the ontology that a dialogue turn refers to




14. Name and explain three challenges in NLU.

1. Spontaneous speech contains mistakes!
   - non-grammatical
   - disfluencies: hesitations, incomplete utterances, self-correction
   - higher ASR errors compared to read speech
2. Possibly unlimited ways to express the same meaning
3. User may behave unexpectedly
   - out-of-domain reply?
   - switching to chat-oriented dialogue?

15. Name and explain two methods for using ASR hypotheses for the NLU inputs. Explain why one is more robust than the other. Explain the ways for NLU prediction using ASR hypotheses.

- **top ASR hypothesis**: Features are extracted directly from single best hypothesis and the classification is performed into relevant semantic classes.
- **N-best list of ASR hypothesis**: This method uses multiple ASR hypotheses to provide a more comprehensive input for NLU.
- N-best list of ASR hypothesis is **more robust** because it incorporates uncertainty in ASR output, reducing the chance of NLU errors.
- **NLU prediction using ASR hypotheses**: the goal is to obtain p(d|a), which is the probability of a dialogue act d given the audio signal a.

16. Give an example of a word confusion network for ASR and explain its benefit.

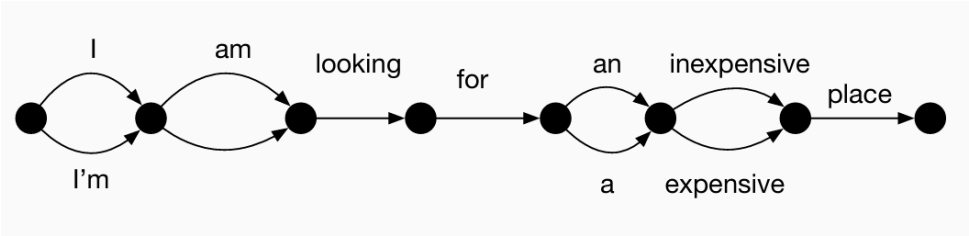



Figure 2: A word confusion network example.

- **Benefit**: It allows NLU systems to consider multiple interpretations of the input, improving robustness against ASR errors.

17. Explain how NLU can be formulated as a sequence-to-sequence learning task.

- NLU can be formulated as a sequence-to-sequence learning task by treating the input utterance as a sequence of tokens and the output as a sequence of semantic representations, such as dialog acts or slot-value pairs.

18. What is MLM (Masked Language Modeling)?

- **Masked Language Modeling** is a self-supervised learning technique where certain tokens in a sentence are masked, and the model learns to predict them based on the surrounding context.

19. Name two challenges for modular dialog system architecture and explain the hybrid approach as its solution.

1. **Challenges**:
   - Modular approaches suffer from information loss between the components.
   - Labeled data not always available to train individual modules.
2. **Hybrid approach**:
   - Combines modular and end-to-end approaches to leverage the strengths of both.
   - Allows for flexibility in component design while ensuring overall system coherence.

20. What is delexicalization, and why is it useful?

- **Delexicalization** is the process of replacing specific values in a dialog system with placeholders, allowing the system to generalize across different instances of the same dialog act.
- **Usefulness**: It helps solve data sparsity.

21. What are some challenges associated with dialog state tracking? Name at least four.

- **Challenges**:
  - Process long context.
  - Remember past information.
  - Infer/Extract implicit information.
  - Solve coreference.

22. Explain the two main approaches to dialog state tracking.

1. **Picklist-based**: Given full ontology, perform prediction over slot-value pairs. Good performance on small datasets, but difficult to scale.
2. **Span-based**: Directly extract slot values from dialogue context. No need for candidate pairs, but struggles with more subtle cases such as implicit choice.

23. What are the automatic and human metrics that can be used for NLG? (Name three for each and explain.)

- **Automatic metrics**:
  - **BLEU**: is a precision-based metric that compares n-gram overlap between generated text and reference text.
  - **ROUGE**: is a recall-based metric that compares n-gram overlap between generated text and reference text.
  - **BERTScore**: Measures contextual embedding similarity.
- **Human metrics**:
  - **Naturalness**: Evaluates how natural and grammatically correct the generated text is.
  - **Informativeness**: Compare the relevance between the generated text and the input or context.
  - **Coherence**: Measures how logically consistent and well-structured the generated text is.

24. What is the interpretation of perplexity as an NLG metric?

- **Perplexity** is a measure of how well a language model predicts a sample. The lower the perplexity, the less confused the model is in making its predictions

25. What is the main difference between BERTScore and other metrics, such as ROUGE and METEOR, in NLG evaluation?

- **BERTScore** measures the semantic similarity between generated text and reference text by comparing their contextual embeddings
- **ROUGE** and **METEOR** focus on n-gram overlap.

26. Explain three things one must consider when evaluating a dialog system with volunteers.

1. **Volunteer Recruitment**: recruit volunteers from a variety of demographics.
2. **Dialogue Task Design**: must be carefully designed.
3. **Researcher Interference**: Researcher should not interfere with the interaction.

27. Name and explain three evaluation factors to be included in the questionnaire of human evaluators for task-oriented and chat-oriented systems.

- **Task-oriented systems**:
  1. **Repetition**: How repetitive was the system?
  2. **Making sense:**: How often did the system say something which did NOT make sense?
  3. **Fluency**: How grammatical was the language?
- **Chat-oriented systems**:
  1. **Interestingness**: How interesting or boring did you find this conversation?
  2. **Inquisitiveness**: How much did the user try to get to know you?
  3. **Engagement**: How much did you enjoy talking to this user?

28. What are the advantages and disadvantages of crowd-sourcing? Name two for each.

- **Advantages**:
  - Scaling up and speeding up evaluation.
  - Reduced cost.
- **Disadvantages**:
  - Reduced quality of interactions.
  - Difficultly controlling who enters the experiment.

29. How does a data-driven automatic dialog measure work, and what are its problems?

- **Data-driven automatic dialog measure** can be trained from a human-labeled dataset with scores. The model then predict the evaluation score for an unseen dialogue.
- **Problems**: domain-dependence, lack of interpretability, single output, susceptibility to adversarial attacks

30. What are the benefits and drawbacks of using prompting for dialog evaluation?

- **Benefits**:
  - Intuitive, can express more nuanced criteria.
  - Time and cost effective.
  - More and more powerful models now available.
- **Drawbacks**:
  - Result may be model and prompt dependent.
  - LLMs have biases, optimizing for LLM evaluation may create a feedback loop that exacerbates these biases.
  - Data contamination.

31. What are the two main features of an ideal evaluation metric?

1. Automatic and thus repeatable.
2. Correlates highly with human judgement.

32. What is the antibiotic effect in the context of dialog evaluation metrics?

- The **antibiotic effect** is the phenomenon where optimization on one fixed automatic metric can lead to models that exploit weaknesses in the metric rather than truly improving performance.

33. Name and explain the factors that make an NLG system good.

1. **Adequacy**: correct meaning.
2. **Fluency**: linguistic fluency.
3. **Readability**: fluency in dialogue context.
4. **Variation**: multiple realisations for the same concept.

34. What is the pros and cons of a template-based natural language generator?

- **Pros**: simple, usually error free, controllable
- **Cons**: time consuming, rigid, not scalable

35. Name and explain the components of the trainable generator pipeline (by Walker).

1. **Sentence plan generator**:
   - Produces multiple sentence plans for a given dialogue act (or set of dialogue acts) and governs which information should be given in which order.
2. **Sentence plan reranker**:
   - Ranks possible candidates.
3. **Surface realiser**:
   - Turns the top candidate into an utterance.

36. How does class-based language modeling work for NLG?

- **Classes**:
  - *inform_area*
  - *inform_address*
  - *inform_phone*
  - *request_area*
  - ...
- **Generation process**:
  1. Generate utterances by sampling words from a particular class language model in which the dialogue act belongs to.
  2. Re-rank utterances according to scores.

37. How does an RNN network improve NLG over its predecessors? In other words, what are its main features?

- An RNN (Recurrent Neural Network) as a language generator improves over its predecessors due to its natural ability to model sequences.
- Main features:
  - **Handling Long-Term Dependencies**
  - **Flexibility to Condition on Auxiliary Inputs**

38. How does LSTM prevent the vanishing gradient problem?

[ Needs some rework ]

- Consider memory cell, where recurrence actually happens:

$$C_t = i_t \odot \hat{C}_t + f_t \odot C_{t-1}$$

- We can back-propagate the gradient by chain rule:

$$\frac{\partial E_t}{\partial C_{t-1}} = \frac{\partial E_t}{\partial C_t}\frac{\partial C_t}{\partial C_{t-1}} = \left(\frac{\partial E_t}{\partial C_t}\right) f_t$$

39. Why can a semantically conditioned LSTM be better than a non-conditional LSTM?

- A semantically conditioned LSTM can be better than a non-conditional LSTM because it incorporates additional semantic information into the generation process, allowing it to produce more contextually relevant and coherent responses.
- This conditioning helps the model to focus on specific aspects of the dialog state, leading to improved performance in generating appropriate utterances.

40. What distinguishes GPT-3 from earlier GPT models?

- Effectively the same model structure with substantially more parameters and training data:

| | GPT | GPT-2 | GPT-3 |
|---|---|---|---|
| Parameters | 117M | 1.5B | 175B |
| Data | 12GB | 40GB | 570GB |

41. What is a major drawback of GPT, as well as semantically conditioned GPT?

- **Cons**: produces hallucinations

42. What are the three key elements in each dialog turn in dialog management?

1. **Actions**: What the system says.
2. **States**: What the user wants.
3. **Observations**: What the system hears.

43. What are the three main challenges in modeling dialog, and how does defining it as a control problem provide a solution?

1. How to define the state space?
2. How to tractably maintain the dialogue state?
3. Which actions to take?

- **Solution**: Define dialogue as a control problem where the behaviour can be automatically learned.

44. How is dialog management framed in a Markov Decision Process (MDP)?

1. **Data**:
   - Dialogue states
   - Reward: a measure of dialogue quality
2. **Model**:
   - Markov decision process
3. **Predictions**:
   - Optimal system actions

- $s_t$: dialogue states
- $a_t$: system actions
- $r_t$: rewards
- $p(s_{t+1} \mid s_t, a_t)$: transition probability

45. What is the main difference between generative and discriminative models in belief tracking?

- **Discriminative models**: focus on the relationship between observations and states without modeling how the observations were generated.
- **Generative models**: model both the states and the observations, considering how states generate observations. The probability of the state depends on how likely it is that this state generated the given observation, combined with our prior belief about the state itself.

46. Why is exact belief tracking intractable in partially observable MDP-based dialog systems?

- Requires sum over all possible states at every dialogue turn - **intractable**!

47. Name and explain three requirements for belief tracking.

1. **Dialogue history**: The system needs to keep track of what happened so far in the dialogue. This is normally done via the **Markov property**.
2. **Task-orientated dialogue**: The system needs to know what the user wants. This is modelled via the **user goal**.
3. **Robustness to errors**: The system needs to know what the user says. This is modelled via the **user act**.

48. What is the Hidden Information State (HIS) model in dialog systems, and how does it address the challenges of belief tracking in partially observable environments?

- The Hidden Information State (HIS) model is a practical framework for POMDP-based spoken dialogue management.
- It addresses the challenges of belief tracking in partially observable environments by structuring the belief state around several components:
  1. **Observation**: N-best list of user acts
  2. **User Goal**: Partitions of the goal space built according to ontology
  3. **Dialogue history**: Grounding states
  4. **Hypotheses**: Every combination of user act, partition and history
- **Belief state**: Distribution over most likely hypotheses

49. What is the Bayesian Update of Dialogue State (BUDS) model, and how does it improve upon previous approaches like the Hidden Information State (HIS) model in belief tracking?

- The Bayesian Update of Dialogue State (BUDS) model is an advancement in POMDP frameworks for spoken dialogue systems.
- Improve upon previous approaches by:
  - Further decomposes the dialogue state
  - Produces tractable belief state update
  - Transition and observation probability distributions can be parametrized and their shape learned

50. Why is grounding important in open-domain dialog systems?

- Prevent hallucinations and improve factual accuracy
- Improve trustworthiness and transparency
- Respond accurately on dynamic or rare topics

51. What is Retrieval-Augmented Generation (RAG), and how does it work? Name and explain its two main components.

- Retrieval-Augmented Generation (RAG) is a framework that enhances generation models with real-time document retrieval.
- Main components:
  1. **Retriever**: Finds relevant documents based on input query
  2. **Generator**: Produces the final response using retrieved docs

52. What is the key advantage of separating the retriever from the generator in RAG systems?

- Up-to-date knowledge without retraining

53. Name two advantages of RAG systems over traditional end-to-end language models.

1. Fewer hallucinations with citation-worthy sources
2. Modular: swap retriever/generator independently

54. What are the four pipeline stages in a RAG system?

1. Encode user input to query embedding
2. Retrieve top-k documents (using DPR, BM25, etc.)
3. Concatenate each document with query
4. Generate response using encoder-decoder model (BART, T5)

55. What are three limitations of RAG systems?

1. Slow response due to retrieval latency
2. Noisy or irrelevant documents degrade output
3. Generator might ignore retrieved text

56. How can retriever quality be improved in RAG systems?

- Use dense retrievers (e.g., DPR, ColBERT) over sparse (e.g. BM25)
- Fine-tune retriever on dialog queries
- Rerank candidates with cross-encoder models

57. What are two techniques to improve alignment between the generator and retrieved knowledge?

1. Train with gold-grounded responses
2. Use contrastive learning between relevant and irrelevant docs
3. Penalize hallucinations with RLHF or fact-checking signals

58. What are the three key evaluation metrics for grounded dialog?

1. **Faithfulness**: Is it factually accurate?
2. **Relevance**: Is it context-appropriate?
3. **Attribution**: Does it cite or reflect the retrieved source?

59. What is the potential risk of RAG systems in open-domain dialog?

- **Leakage of private data**, if the external datastore used by the RAG system contains sensitive or private information, there is a risk that this data could be retrieved and inadvertently exposed in the system's responses.
- Could still produce **hallucinations or misleading information** if the retrieved data itself is incorrect, biased, or misinterpreted by the language model.

60. Name and explain three reasons why open-domain dialog is more challenging than task-oriented dialog.

1. **Unlimited topics**: Users can ask about anything
2. **Ambiguity**: Same phrase can have many meanings
3. **Context management**: Conversations span multiple turns
4. **Lack of grounding**: Risk of hallucinating information

61. Why are generic responses like "I don't know" problematic in chatbot conversations, and how can they be mitigated?

- **Problem**: Safe defaults like "I don't know" or "That's interesting."
  - Reduce user engagement
  - Perceived as evasive or boring
- **Mitigation**:
  - Penalize frequent responses in decoding (frequency penalties)
  - Use top-k/top-p sampling to promote lexical diversity
  - Train with contrastive or adversarial losses for engaging answers

62. What are three ways to mitigate hallucination in dialog systems?

1. Incorporate retrieval-based grounding (RAG)
2. Post-generation fact checking modules
3. Human feedback training to penalize factual errors

63. What is a major risk when a chatbot loses context in multi-turn conversations, and what are two ways to mitigate it?

- **Problem**: Bots lose track of user preferences or dialog history
  - Responses feel disconnected
  - Violates consistency, especially for multi-turn dialog
- **Mitigation**:
  - Use long-context transformers (e.g. Longformer, GPT with memory)
  - Add dialog memory modules
  - Use dialog history explicitly as input

64. What types of errors fall under pragmatic errors in chatbot responses, and what are three ways to mitigate such errors?

- **Problem**: Bot says things that are socially or contextually inappropriate
  - Fails to detect sarcasm, humor, formality
  - Lacks understanding of politeness or indirectness
- **Mitigation**:
  - Train on datasets annotated for tone, register
  - Use style-conditioned generation
  - Explicit modeling of Grice's Maxims

65. How can chatbots mitigate toxic or biased responses?

- Filter training data using toxicity classifiers
- Post-process outputs with safety layers
- Reinforcement learning from human feedback (RLHF)

66. Why is it difficult for chatbots to handle multi-intent utterances like 'Book a flight to Tokyo and what's the weather there?'

- **Problem**: Users combine multiple intents in one utterance
- **Example**: "Book me a flight to Tokyo and what's the weather there?"
  - Requires multi-task understanding
  - Potentially separate dialog states
- **Mitigation**:
  - Semantic parsing into sub-tasks
  - Intent detection + follow-up prompts

67. What distinguishes a retrieval-based language model from a standard language model?

- It retrieves from an external datastore (at least during inference time)

68. Give five reasons why we need retrieval-based language models.

1. LLMs can't memorize all (long-tail) knowledge in their parameters
2. LLMs' knowledge is easily outdated and hard to update
3. LLMs' output is challenging to interpret and verify
4. LLMs are shown to easily leak private training data
5. LLMs are **large** and expensive to train and run

69. What are the three main design questions for retrieval-based language models?

| **1. What to retrieve?** | **2. How to use retrieval?** | **3. When to retrieve?** |
|---|---|---|
| • Chunks | • Input layer | • Once |
| • Tokens | • Intermediate layers | • Every n tokens (n>1) |
| • Others | • Output layer | • Every token |

70. What is the main idea behind kNN-LM?

- A different way of using retrieval, where the LM outputs a nonparametric distribution over every token in the data.
- Can be seen as an incorporation in the "output" layer

71. Give an advantage and a disadvantage of fine-tuning in adapting an LM to downstream tasks.

- **Advantages**:
  - Customizable
  - Competitive w/ more data
- **Disadvantages**:
  - Requiring training

72. Give an advantage and a disadvantage of reinforcement learning in adapting an LM to downstream tasks.

- **Advantages**:
  - Better alignment with user preferences
- **Disadvantages**:
  - Requiring additional data collection (preference)

73. Give an advantage and a disadvantage of retrieval-based prompting.

- **Advantages**:
  - No training & strong performance
- **Disadvantages**:
  - Hard to control, underperforming full FT model

74. Name and explain three methods for adapting a retrieval-based LM for downstream tasks.

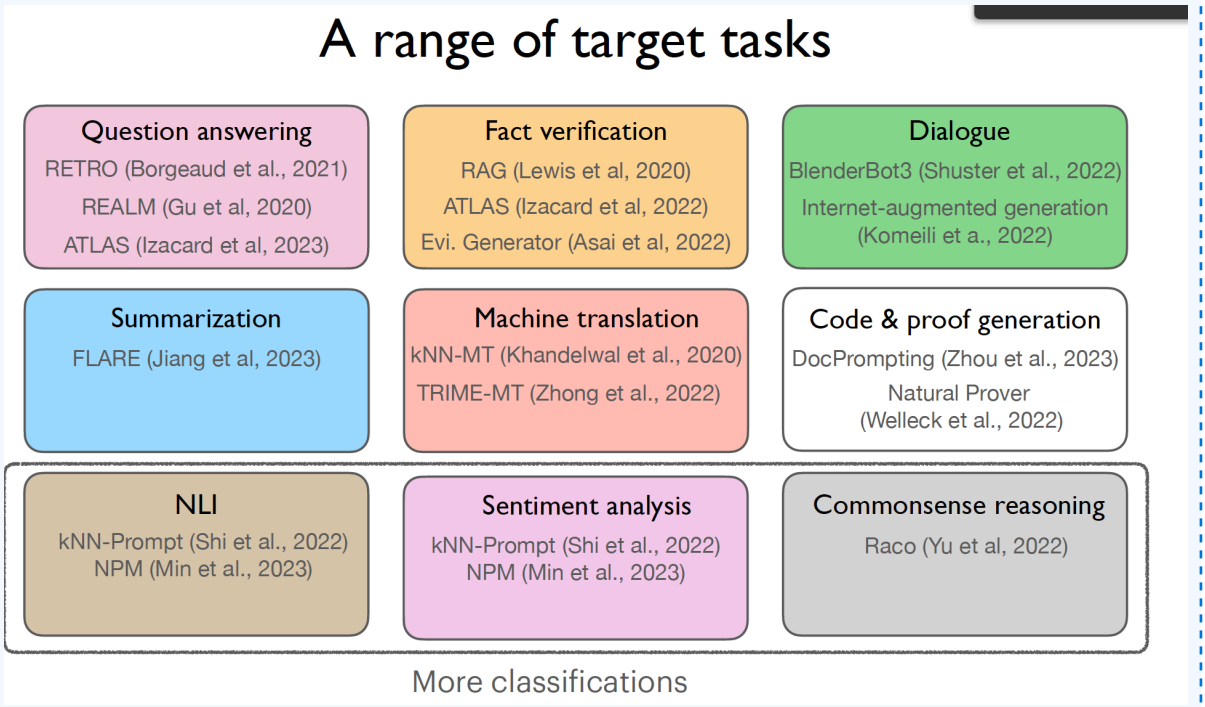
1. **Fine-tuning**:
   - Involves **training the LM and/or the retriever on task-specific data and a datastore**. Fine-tuning can be done on both the retriever and the LM, or just the query-side of the retriever while fixing the index.
2. **Reinforcement learning**:
   - Uses **human feedback to optimize the language model's policy**. Instead of relying solely on supervised fine-tuning, RL allows the model to learn to present information and align with human preferences for aspects like helpfulness, honesty, and harmlessness.
3. **Prompting**:
   - Involves **providing retrieved knowledge as part of the input context to a frozen (pre-trained) LM**. It does not require additional training of the LM on downstream tasks.

75. Name and explain four key effectiveness points in downstream tasks for retrieval-based LMs.

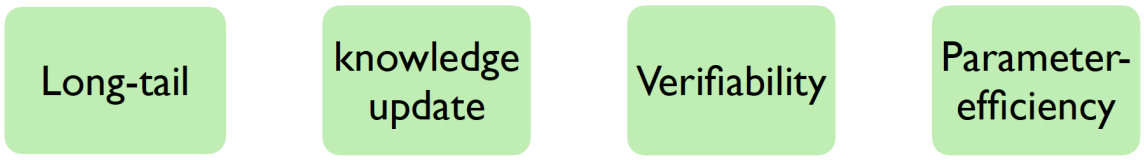
1. **Long-tail**:
   - LLMs often struggle in **long-tail/less frequent entities**. Scaling LLMs only helps for **popular knowledge**; for long tail, scaling gives marginal performance improvements Retrieval gives large performance gain in such **long-tail**. Largely reduce hallucinations in **long-form generations**.
2. **Knowledge update**:
   - Standard LLMs need to be **trained again** to adapt to evolving world knowledge. Swapping the knowledge corpus to **accommodate temporal changes** without additional training.
3. **Verifiability**:
   - **Much smaller LMs with retrieval** can outperform much larger LMs in fact completions.
4. **Parameter-efficiency**:
   - Human and model can reliably assess the **factuality of the generations** using the retrieved evidence.

76. Name and explain five scenarios when retrieval-based LMs should be used.

scenarios




why




77. What is an agent?

- LLM-powered Agents are artificial entities that enhance LLMs with essential capabilities enabling them to sense their environment, make decisions, and take actions.

- An "intelligent" system that interacts with some "environment"
  - Physical environments: robot, autonomous car, …
  - Digital environments: DQN for Atari, Siri, AlphaGo
  - Humans as environment: Chatbots

78. Name and explain the two competing views on agents.

## Two competing views

**LLM-first view**: We make an LLM into an agent!

- Implications: scaffold on top of LLMs, prompting-focused, heavy on engineering

**Agent-first view**: We integrate LLMs into AI agents so they can use language for reasoning and communication!

- Implications: All the same challenges faced by previous AI agents (e.g., perception, reasoning, world models, planning) still remain, but we need to **re-examine them through the new lens of LLMs** and tackle new ones (e.g., synthetic data, self-reflection, internalized search)

79. What is the fundamental difference between current and classic agents?

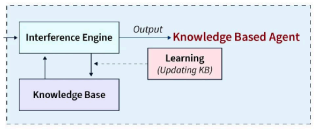
## What's fundamentally different now?

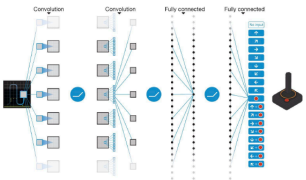
Contemporary AI agents, with integrated LLM(s), can *use language as a vehicle for reasoning and communication*

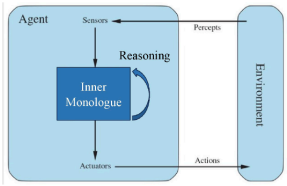
- ⬆ Instruction following, in-context learning, output customization
- ⬆ Reasoning (for better acting): state inferences, self-reflection, replanning, etc.

80. Name three types of AI agents and compare them in terms of expressiveness, reasoning and adaptivity.

## Evolution of AI agents




**Logical Agent** | **Neural Agent** | **Language Agent**

| | | | |
|---|---|---|---|
| **Expressiveness** | Low<br>bounded by the logical language | Medium<br>anything a (small) NN can encode | High<br>almost anything, esp. verbalizable parts of the world |
| **Reasoning** | Logical inferences<br>sound, explicit, rigid | Parametric inferences<br>stochastic, implicit, rigid | Language-based inferences<br>fuzzy, semi-explicit, flexible |
| **Adaptivity** | Low<br>bounded by knowledge curation | Medium<br>data-driven but sample inefficient | High<br>strong prior from LLMs + language use |

81. Name and explain three methods that can be used to teach LLMs how to properly use tools.

- Tutorial Learning
  - Have model tuned for tool use read tool manuals (tutorials), so that it understands the functions of the tool and how to invoke them
  - Works well with powerful LLMs
- Reinforcement Learning
  - Autonomous exploration and correction of errors based on environmental feedback through reinforcement learning
  - Action space defined by tools
  - Agent learns to select appropriate tool
  - Correct action maximize reward signal
- Self-supervised Tool Learning
  - Pre-defined tool APIs
  - Encourage models to call and execute tool APIs
  - Design self-supervised loss to evaluate tool execution helpfulness

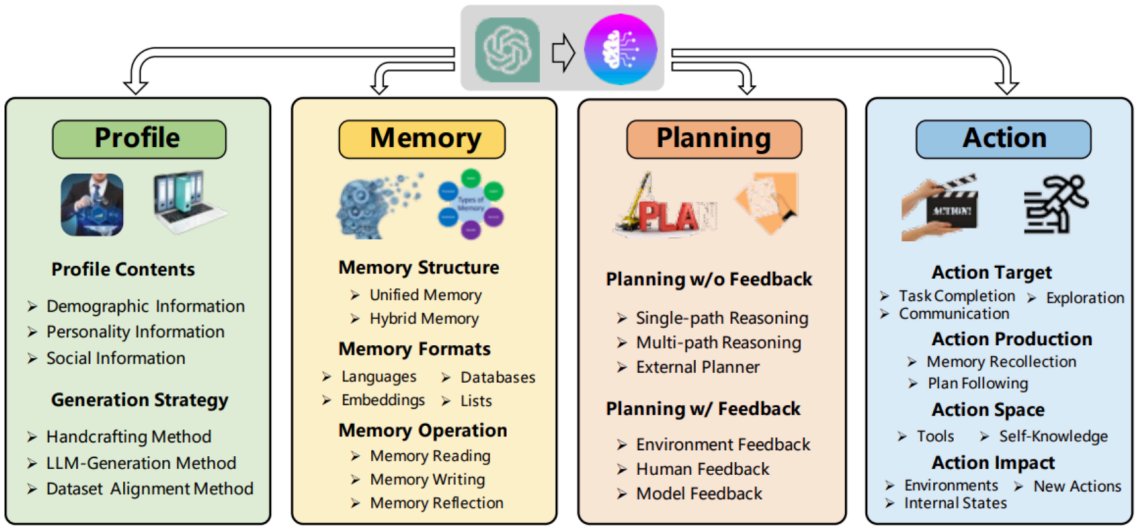
82. What is the ReAct agent, and what are its benefits?

**ReAct**: action space $\hat{A} = A \cup \mathscr{L}$ augmented by reasoning

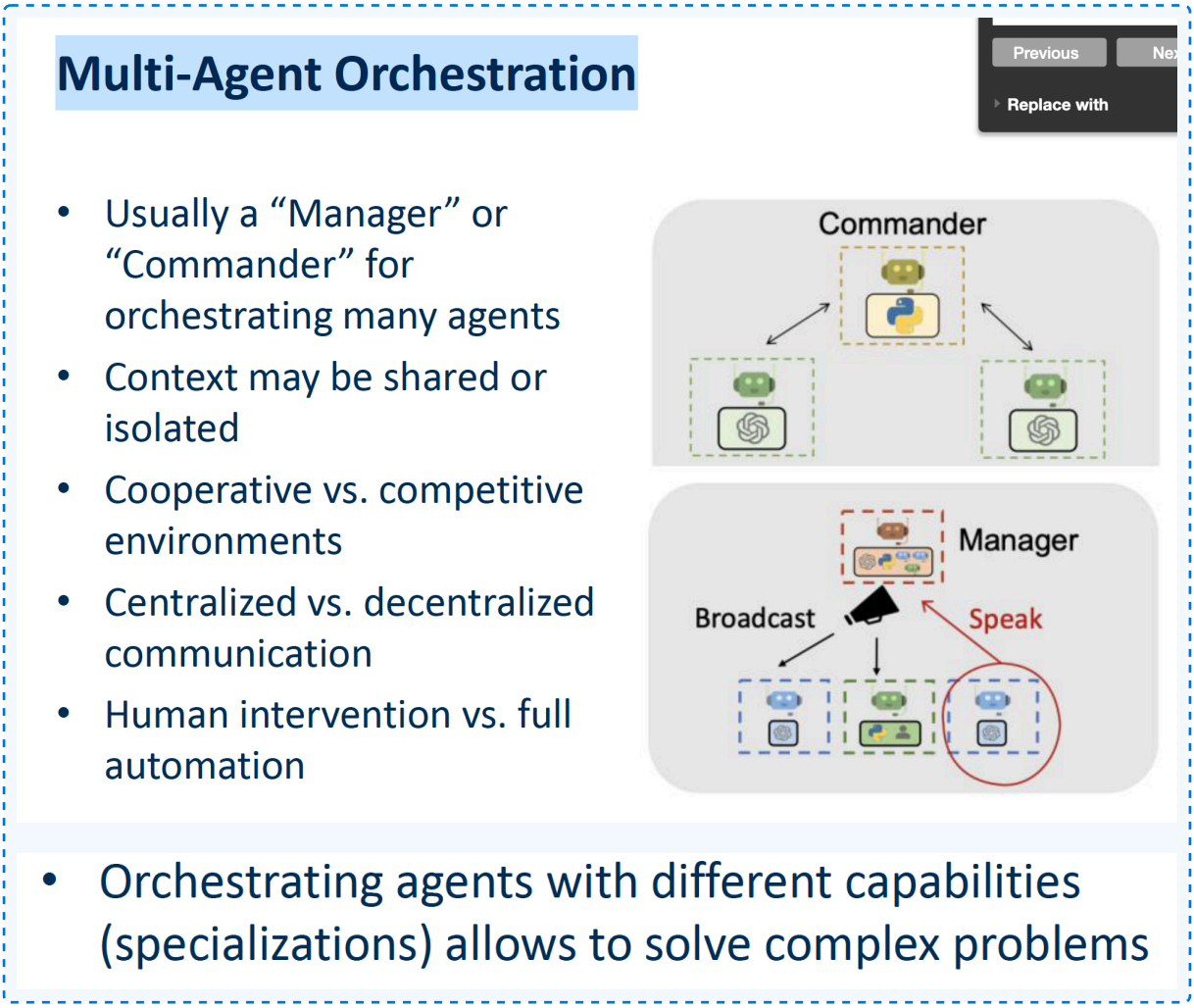
**Reasoning guides acting**

**Reasoning helps diagnose and control acting**

83. What are the four components of the unified framework for LLM agents?

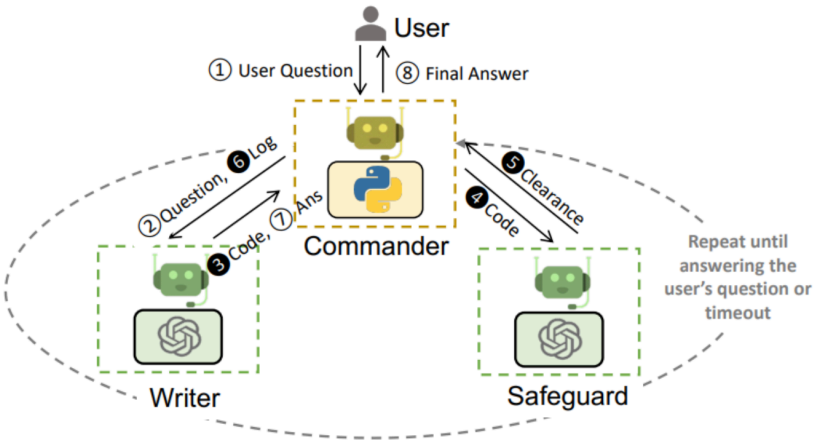



84. What is multi-agent orchestration, and why do we need it?



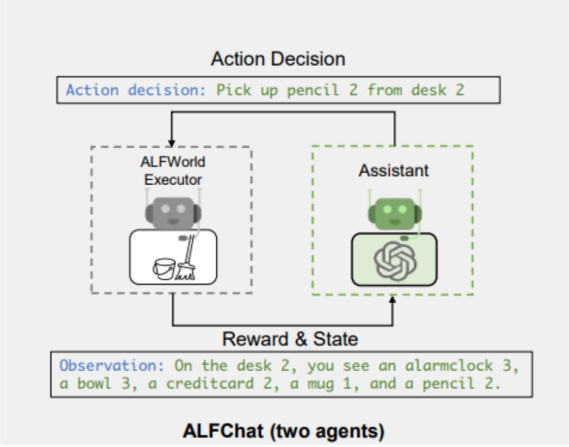

85. Give two examples of a multi-agent system and explain how they work.

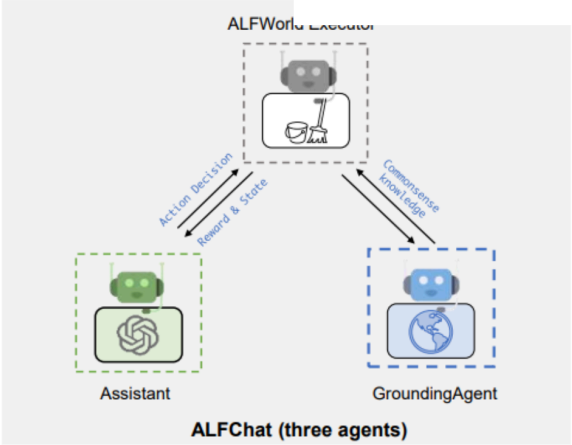
## Example: Multi-Agent Coding




- Commander receives user questions and executes code
- Writer writes code
- Safeguard ensures no information leakage or malicious code

## Example: Decision Making




- Two agents: One suggests next step, Executor does action and provides feedback
- Three agents: additional agent that provides commonsense facts about the domain when needed

86. What are two potential risks of multi-agent systems, and why are they difficult to deal with?

1. Leaking private data
2. Causing financial loss.

- They difficult to deal with because Identifying these risks is labor-intensive as testing becomes difficult with increased agent complexity.

87. What is LLM alignment, and why is it important?

- **LLM alignment** is a learning phase where they learn how to present information to users and align to human preferences.
- It is important because it helps prevent harmful or unintended consequences, ensuring that LLMs act in ways that are beneficial and safe for users.

88. How does Proximal Policy Optimization (PPO) work, and what are its two benefits?

- **Proximal Policy Optimization (PPO)** is a reinforcement learning algorithm that optimizes the policy of an agent by balancing exploration and exploitation.
- **Benefits**:
  1. Prevents mode collapse to single high reward answers.
  2. Prevents the model from deviating too far from the distribution where the reward model is accurate.

89. What are three drawbacks of PPO?

1. Need to train multiple models.
2. Needs sampling from Language model during fine-tuning.
3. Complicated reinforcement learning training process.

90. How does Direction Preference Optimization (DPO) improve upon PPO?

**Direction Preference Optimization (DPO)** improves upon PPO by directly optimizing the policy based on preference data, rather than relying on a reward model.

91. What is Reinforcement Learning from Human Feedback (RLHF)?

**Reinforcement Learning from Human Feedback (RLHF)** is a method that combines reinforcement learning with human feedback to train language models, allowing them to align with human preferences and improve their performance in generating responses.

92. What are the challenges of the *human feedback* in RLHF?

- Biases of human evaluators.
- Good oversight is difficult.
- Data Quality: Cost/Quality trade-off.
- Tradeoff between richness and efficiency of feedback types

93. What are the challenges of the *reward model* in RLHF?

- A single reward model cannot represent a diverse society of humans.
- Reward misgeneralization: reward model may fit with human preference data due to unexpected features.
- Evaluation of reward model is difficult and expensive.

94. What are the challenges of the *policy* in RLHF?

- Robust reinforcement learning is difficult.
- Policy misgeneralization: training and deployment environments are different.

95. Name and explain the three key concepts for language agents.

1. **Reasoning**: The ability to update short-term memory.
2. **Memory**: The capacity to store and retrieve information.
3. **Planning**: The algorithm to choose an action from the action space.

96. Why are reasoning and acting helpful for agents?

- **Reasoning** helps agents to make informed decisions based on the current context and available information.
- **Acting** allows agents to take actions that can change the environment or achieve specific goals, enabling them to interact effectively with users and systems.

97. How do LLM agents have short-term and long-term memories, and what are they most useful for?

- **Short-term memory**: This is primarily the context window of the LLM.
  - Most useful for reasoning.
- **Long-term memory**: retains knowledge and experiences over time, enabling the agent to learn from past interactions and improve future performance.
  - Most usefull for retrieving and learning.

98. Name and explain three planning paradigms for language agents. Give an advantage and a drawback for each.

1. **Reactive**: Combines reasoning and acting in a single framework.
   - **Advantage**: fast, easy to implement.
   - **Drawback**: greedy, short-sighted.
2. **Tree Search with Real Interactions**: Breaks down complex tasks into smaller, manageable steps.
   - **Advantage**: systematic exploration.
   - **Drawback**: irreversible actions, unsafe, slow.
3. **Model-Based Planning**: Organizes actions into a hierarchy of goals and sub-goals.
   - **Advantage**: faster, safer, systematic exploration.
   - **Drawback**: relies on the accuracy and generalizability of the world model.

99. What are the advantages of code agents?

- **Object Management**: They can handle complex data structures (images, audio, texts).
- **Composability**: They allow for the combination and reuse of actions.
- **Generality**: Code agents are capable of performing any computationally possible task.

100. What is the purpose of the Model Context Protocol (MCP)?

The purpose of the Model Context Protocol (MCP) is to dramatically reduce integration complexity and maintenance burden when building AI applications that interact with various tools and data sources. It achieves this by:

- Defining a standard protocol where each AI application implements the client side of MCP only once.
- Requiring each tool or data source to implement the server side of MCP only once.