CS224v

Conversational Virtual Assistants
with Deep Learning

Lecture 4:
Evaluation of Task-Oriented Agents

Monica Lam and Harshit Joshi

Lecture Goals

- Recap of last lecture
- Evaluating Genie Agents

Task Agent Architectures

1. Dialogue Tree:
   - Hard-code statements: users are given a few choices

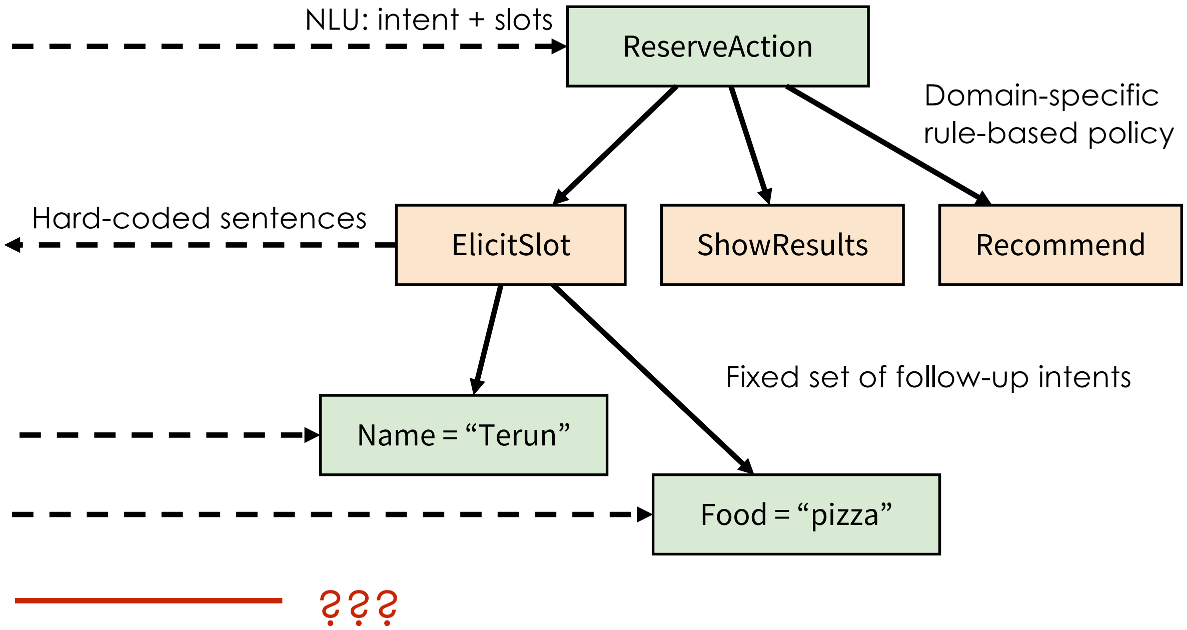
A Restaurant Reservation Agent Dialogue Tree

A: Hello, how can I help you?

U: I'm looking to book a restaurant
for Valentine's Day

A: What kind of restaurant?

U: Terun on California Ave
-- or –
U: Something that has pizza
-- or –
U: I don't know, what do you
recommend?



NLU: intent + slots

ReserveAction

Domain-specific
rule-based policy

Hard-coded sentences

ElicitSlot

ShowResults

Recommend

Fixed set of follow-up intents

Name = "Terun"

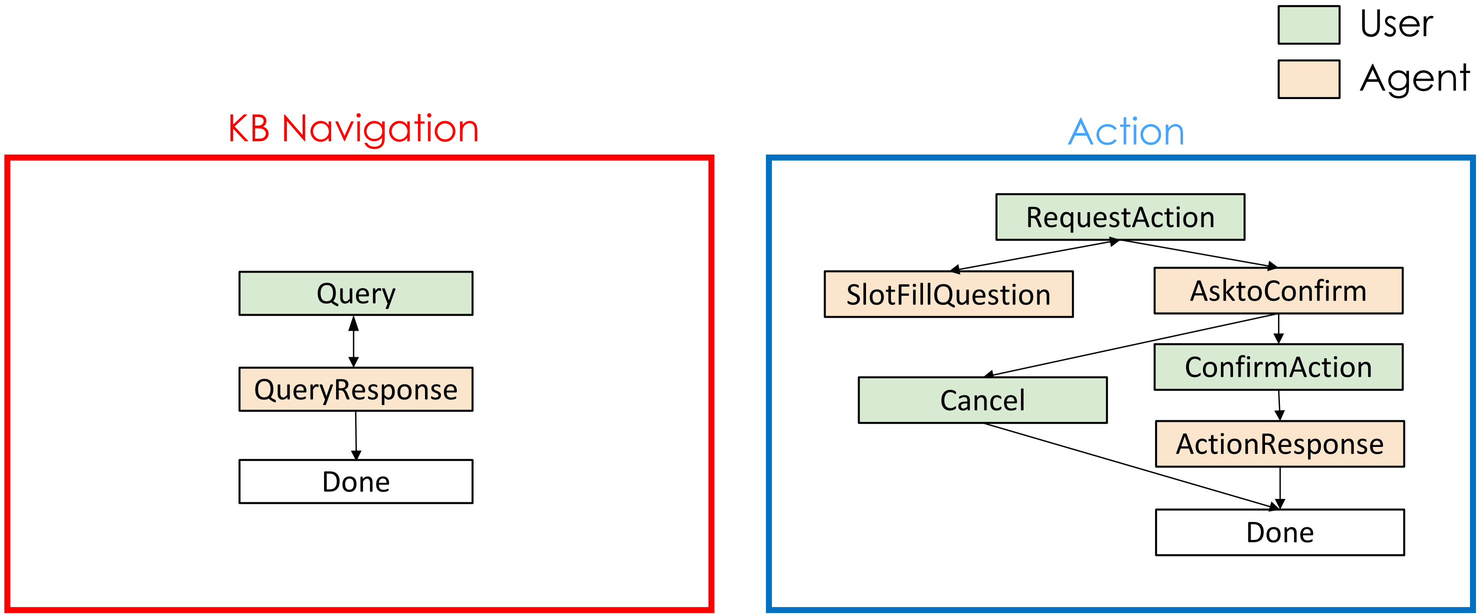
Food = "pizza"

???

Task Agent Architectures

1. Dialogue Tree:
   - Hard-code statements: users are given a few choices
   - Quiz: what the limitations?
2. Finite state machine of dialogue acts (intents):
   - Statements are “factored” into dialogue acts + parameters because similar transitions are taken for the same intent

Example of a Dialogue State Machine



🟩 User

🟧 Agent

KB Navigation

Query

QueryResponse

Done

Action

RequestAction

SlotFillQuestion

AsktoConfirm

ConfirmAction

Cancel

ActionResponse

Done

Dialogue act = Intent:

Independent of the conversation domain

Parameter values are domain-specific

Quiz: what are examples of user/agent dialogue acts

Task Agent Architectures

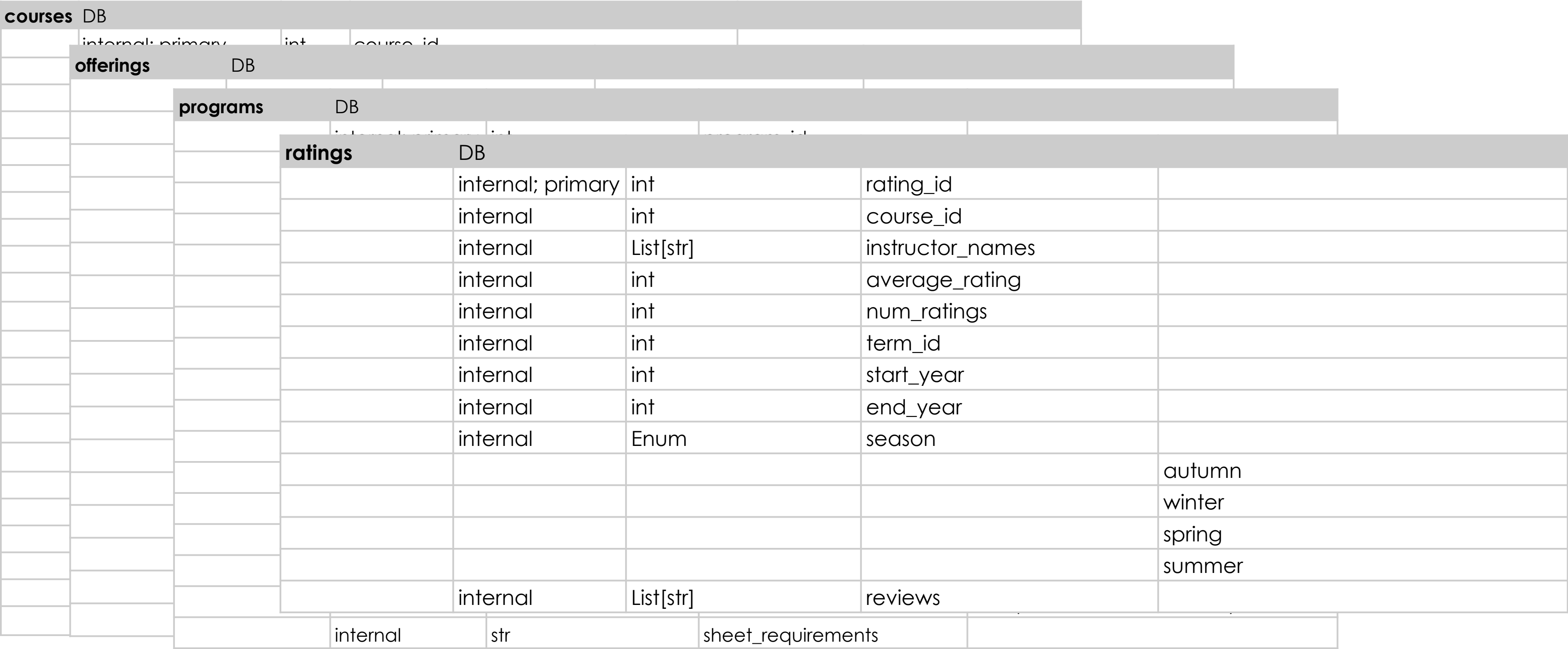
1. Dialogue Tree:
   - Hard-code statements: anticipate possible user statements
   - Quiz: what the limitations?
2. Dialogue state machine with dialogue acts (intents):
   - Statements are "factored" into dialogue acts + parameters because similar transitions are taken for the same intent
   - Quiz: what are the limitations?
3. LLM agents via prompting or fine-tuning
   - Quiz: What are the limitations?

Genie Worksheet

- Principles:
  - **Unlike LLMs**: the agent policy needs to be **controlled**
  - **Unlike Dialogue State Machines**: Needs to be more **flexible**

- DECLARATIVE specification
  - **Worksheets**: Like forms that you fill on the website
    - Variables: info to fill in
    - Actions: rules on what the agent should do given the values
  - **Knowledge bases**: Answer any question on the database
    - By translating natural language into formal queries

1. Course Assistant Knowledge Corpus



courses DB

	internal; primary	int	course_id	

offerings DB

programs DB

	internal; primar	int	program_id	
	internal	str	sheet_requirements	

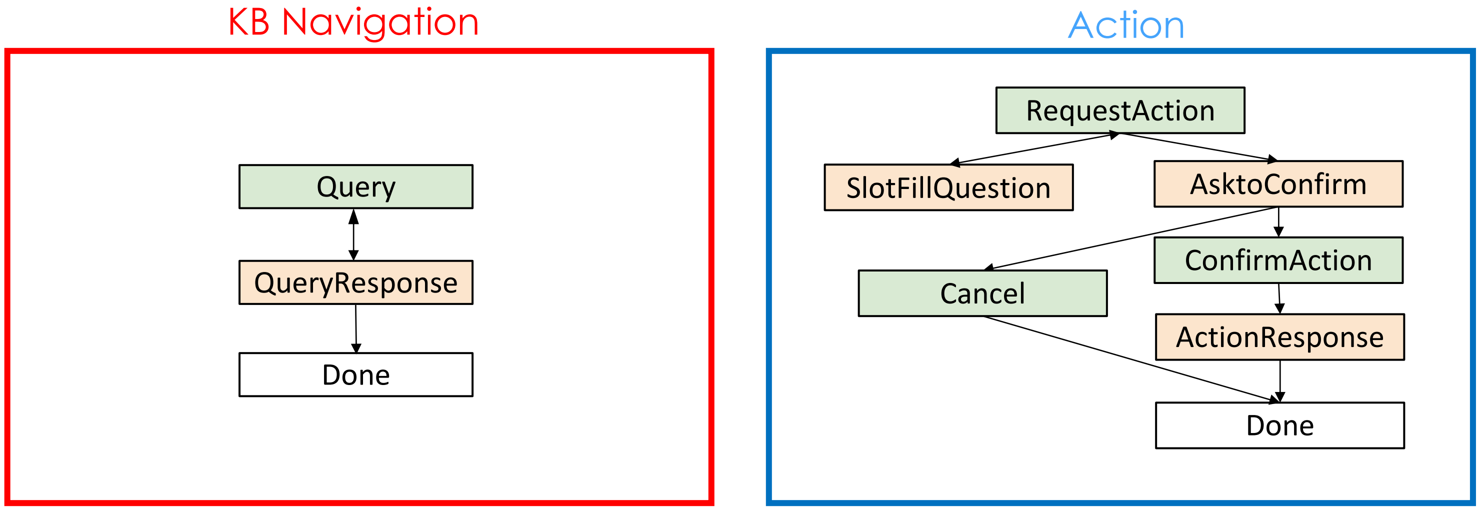
ratings	DB			
	internal; primary	int	rating_id	
	internal	int	course_id	
	internal	List[str]	instructor_names	
	internal	int	average_rating	
	internal	int	num_ratings	
	internal	int	term_id	
	internal	int	start_year	
	internal	int	end_year	
	internal	Enum	season	
				autumn
				winter
				spring
				summer
	internal	List[str]	reviews	

2. Stanford Course Enrollment Form

Form Name	Kind	Type	Name	Enum Values	Description	WS Action
Main	WS		course_enrollment			Say(enroll(StudentInfo, courseToTake))
	input	CourseToTake	course_to_take		The course to enroll	
	input	StudentInfo	student_info_details		Information on the student	
StudentInfo		worksheet				
	input	str	student_name		Name of the student	
	input	str	student_id		Student's ID number	
	input	str	student_email_address		Student's email address	
CourseToTake		worksheet				
	input	str	course_name		Name of the course	
	input	Enum	grade_type		The desired grading basis	
				Cred/No Cred		
				Letter		
	input	int	num_units		The number of units taken	
	input	confirm	confirm		Confirm the course	

Comparison with Dialogue State Machines

🟩 User
🟧 Agent



Dialogue act = Intent:

Independent of the conversation domain
Parameter values are domain-specific

Quiz: what is the difference between the Dialogue State Machine vs. Genie Worksheets?

Genie Worksheet (WS) vs. Dialogue State Machine

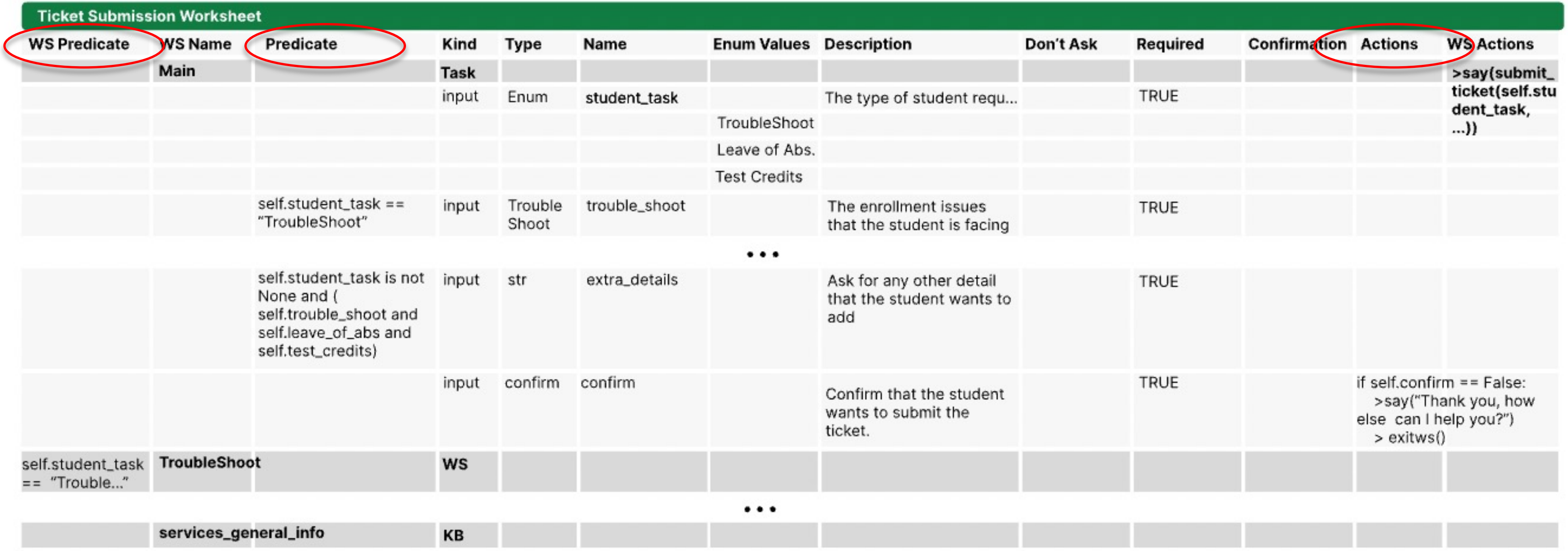
Similarities

- **Declarations of slots, automatic slot filling questions, confirmations**

Differences: **WS run-time is a not a Finite-State Machine**

- **WS supports multiple queries and requests in a single utterance**
- **WS supports arbitrary interleaving of queries & action requests**
  - Agents' queries may not be answered right away!
  - User can pose a question any time
  - The agent needs to go back to outstanding requests
  - Quiz: Can we use a stack so we can return to outstanding requests?
  - The user can change any field value any time
    - Genie run-time scans the WS to process outstanding requests

+ Data-Dependent Worksheets/Fields
+ Field actions

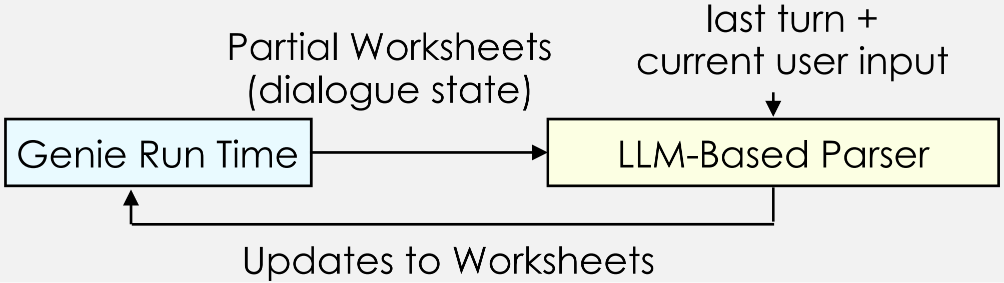


Ticket Submission Worksheet

WS Predicate	WS Name	Predicate	Kind	Type	Name	Enum Values	Description	Don't Ask	Required	Confirmation	Actions	WS Actions
	Main		**Task**									**>say(submit_ticket(self.student_task, ...))**
			input	Enum	student_task		The type of student requ...		TRUE			
						TroubleShoot						
						Leave of Abs.						
						Test Credits						
		self.student_task == "TroubleShoot"	input	Trouble Shoot	trouble_shoot		The enrollment issues that the student is facing		TRUE			
						•••						
		self.student_task is not None and (self.trouble_shoot and self.leave_of_abs and self.test_credits)	input	str	extra_details		Ask for any other detail that the student wants to add		TRUE			
			input	confirm	confirm		Confirm that the student wants to submit the ticket.		TRUE		if self.confirm == False: >say("Thank you, how else can I help you?") > exitws()	
self.student_task == "Trouble..."	**TroubleShoot**		**WS**									
						•••						
	services_general_info		**KB**									

How Genie Handles Long Conversations

INPUT



Partial Worksheets (dialogue state)

last turn + current user input

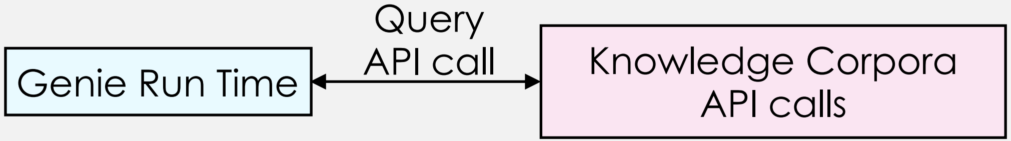
Genie Run Time

LLM-Based Parser

Updates to Worksheets

EXECUTE POLICY

- Queries
- Activated actions



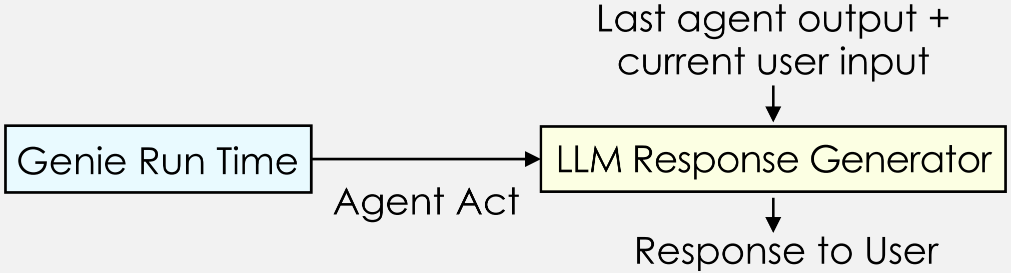
Genie Run Time

Query API call

Knowledge Corpora API calls

OUTPUT

- Responses from above
- Asks for confirmation
- Asks for missing parameters in the worksheet



Last agent output + current user input

Genie Run Time

Agent Act

LLM Response Generator

Response to User

Summary: Genie Worksheets Can

- Support multiple queries and requests in a single utterance
- Support arbitrary interleaving of queries & action requests
- Support data-dependent parameter queries
- Keep track of essential information in long conversations
- Allow unhappy paths, eg. User changing their answers

Developer only needs to supply the Genie Worksheet

Quiz: Can LLMs handle this level of complexity?

Evaluation

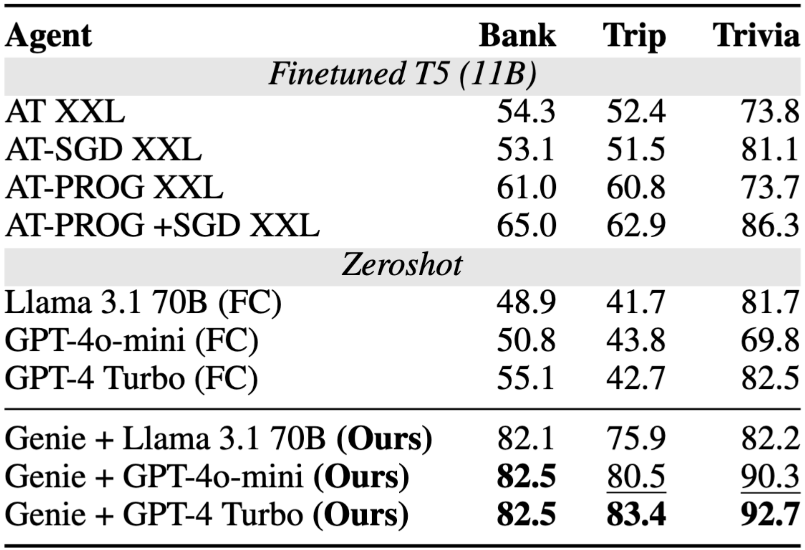
- To compare with existing work: STARV2 Dataset (2022)
  - Wizard-of-Oz: 3 hardest domains--bank, trivia, trip
  - Annotated with: “user and agent dialogue acts”

Mosig, Johannes EM, Shikib Mehri, and Thomas Kober. "Star: A schema-guided dialog dataset for transfer learning.”
Zhao, Jeffrey, et al. "Anytod: A programmable task-oriented dialog system."

STARv2 Evaluation

- SOTA: AnyTOD (**AT XXL**)
  - finetuned T5 (13B) on 6000 examples (all domains but the test)
  - Implement programs to handle logic
- Genie:
  - Semantic parsing prompt: 3 examples
  - Agent Policy: 9 lines in a Worksheet

System Action F1: The next agent action



Agent	**Bank**	**Trip**	**Trivia**
Finetuned T5 (11B)			
AT XXL	54.3	52.4	73.8
AT-SGD XXL	53.1	51.5	81.1
AT-PROG XXL	61.0	60.8	73.7
AT-PROG +SGD XXL	65.0	62.9	86.3
Zeroshot			
Llama 3.1 70B (FC)	48.9	41.7	81.7
GPT-4o-mini (FC)	50.8	43.8	69.8
GPT-4 Turbo (FC)	55.1	42.7	82.5
Genie + Llama 3.1 70B (**Ours**)	82.1	75.9	82.2
Genie + GPT-4o-mini (**Ours**)	**82.5**	80.5	90.3
Genie + GPT-4 Turbo (**Ours**)	**82.5**	**83.4**	**92.7**

Most errors caused by inconsistent data annotations

Evaluation

Two-part evaluation

- To compare with existing work: STARV2 Dataset (2022)
  - Wizard-of-Oz: 3 hardest domains--bank, trivia, trip
  - Annotated with: "user and agent dialogue acts"
  - WOZ benchmarks:
    - not realistic -- few unhappy paths
    - too easy for LLMs (simple slot fills)
    - Worse annotations than LLM outputs!
- Real use cases with real users!

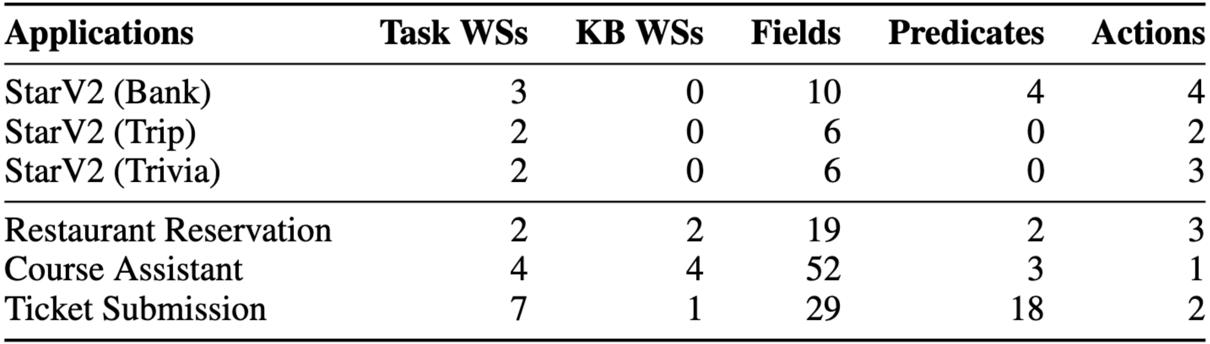
LLM out-ran traditional NLP evaluation!

Mosig, Johannes EM, Shikib Mehri, and Thomas Kober. "Star: A schema-guided dialog dataset for transfer learning."
Zhao, Jeffrey, et al. "Anytod: A programmable task-oriented dialog system."

Evaluation with real users

Three diverse applications with varying complexities.

- **Restaurant Reservation**: Uses the real-life database from Yelp
- **Ticket Submission**: A subset of Service Now api
- **Course Enrollment**: Uses multiple real-life databases of courses at Stanford University.



Applications	Task WSs	KB WSs	Fields	Predicates	Actions
StarV2 (Bank)	3	0	10	4	4
StarV2 (Trip)	2	0	6	0	2
StarV2 (Trivia)	2	0	6	0	3
Restaurant Reservation	2	2	19	2	3
Course Assistant	4	4	52	3	1
Ticket Submission	7	1	29	18	2

Note the difference in complexity

Baseline and Study Design

- Compare to GPT-4 turbo with Function Calling.
  - Gives GPT-4 the same KB-Parser as Genie Worksheet
- Users
  - Restaurant Reservation and Ticket Submission:
    - 22 and 20 users from Prolific crowdsource platform
  - Course enrollment:
    - 20 university students
  - Randomly assigned Genie Agent or GPT-4 (FC)

Real User Evaluation

Goal Completion Rate

	Restaurant	Course Enrollment	Ticket submission
GPT4: function calling + Genie DB parser	**54.5**	**10.0**	**0.0**
Genie Worksheet	**91.6**	**80.0**	**80.0**

GPT-4 results are not acceptable

- Low completion rate: even 54.5% on restaurant is not good enough!
- Course: Hallucinates non-existent courses despite using knowledge base results

Genie Worksheets – deployable with additional engineering

- WS Only LLM-based technique that does not hallucinate on Query results/Actions
- Pilots ongoing with companies

Homework

- Part 1: Interact with the fidelity investment agent built using Genie Worksheets

- Part 2: Create a ride request agent using Genie Worksheet

Projects

Applications

- AI-Powered Heart Failure Medication Assistant:
  Revolutionizing Patient Care Through Intelligent Conversation

Research: to address the difficulty in creating a complete Genie Worksheet

- **GenieWorksheet Wizard**:
  Discovery of Missing Capabilities in Task-Oriented Agents from Simulation
  (Case study: Fidelity investments)
- **From Real Conversations** to a Complete Genie Worksheet!

Course Participation

- Lectures are interactive
  - You are encouraged to ask questions & answer my quizzes

- Please consider pitching your project on Wed
  and get early feedback and find partners.

For participation
You are not graded on the quality of your questions/answers/pitches