AI for task handover: advances in communication for model reconciliation

Kayleigh (Kaleb) Bishop

Preliminary exam

Intro: What am I talking about?

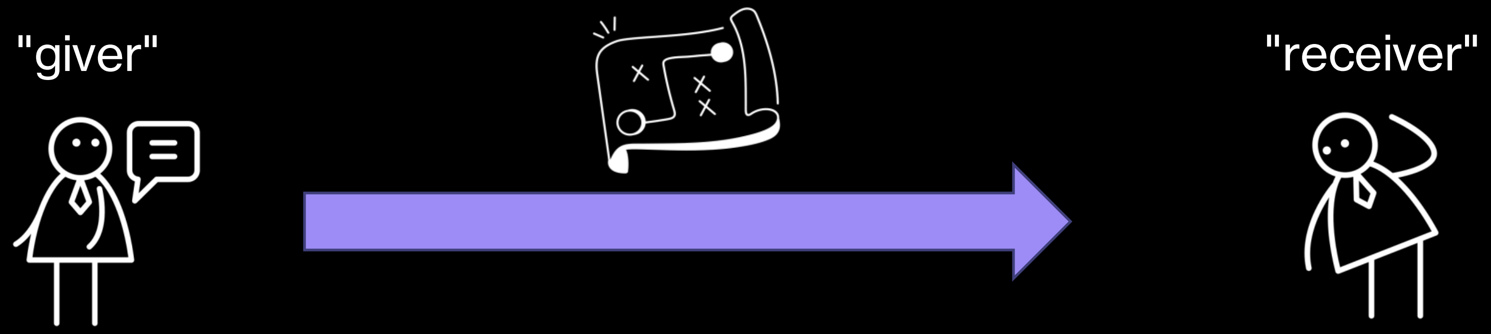
- What is task handover?
- What's so hard about it?
- Why do we care?
- Why get AI involved?

[Image credit: mindtheproduct.com]

What is task handover?

The process of communicating the necessary information to transfer the responsibility for a task from one agent, or team, to another.

Happens regularly in shift work environments: medicine, manufacturing, hospitality...



Motivating example: medicine

- Nurses must handover patient care at end-of-shift
- Relies on combining health records and own experiences – all as it pertains to continuity of care
- Many standardization tools in use to make sure key info is captured, but dialogue and synthesis still required

[Image: IPASS Institute]

Example written handoff

Illness Severity	Patient Summary	Action List	Situation & Contingency	Synthesis by Receiver
Watcher	45 y/o female with Type 1 Diabetes, who was admitted yesterday for DKA and sepsis secondary to a UTI. She was initially admitted to the ICU, where she received an insulin drip, aggressive hydration, and was started on ceftriaxone for her UTI. Insulin drip was stopped overnight when gap closed, started on subcutaneous insulin prior to stopping drip. She was transferred to the medicine floor this AM in stable condition. Glycemic control: - continue subcutaneous insulin: 15U Aspart QAC and 40U Glargine tonight. - Last fingerstick glucose 1hr ago was 233. Urosepsis:	- make sure Glargine is started - follow up 8PM BMP - monitor her respiratory status Q4h - monitor fever cure	- Replete her electrolytes if K <4, Phos <2, and Mg <2. - If anion gap increases insulin drip may need to restart - For resp distress get a CXR and consider Lasix 40 mg IV - If she develops a fever > 101F, please re-culture urine and blood and broaden antibiotics to Zosyn.	An example synthesis will be included in the spoken handoff that follows.

Example written handoff

Illness Severity

Watcher

Patient Summary

45 y/o female with Type 1 Diabetes, who was admitted yesterday for DKA and sepsis secondary to a UTI. She was initially admitted to the ICU, where she received an insulin drip, aggressive hydration, and was started on ceftriaxone for her UTI. Insulin drip was stopped overnight when gap closed, started on subcutaneous insulin prior to stopping drip. She was transferred to the medicine floor this AM in stable condition.

Glycemic control:

- continue subcutaneous insulin: 15U Aspart QAC and 40U Glargine tonight.

- Last fingerstick glucose 1hr ago was 233.

Urosepsis:

Action List

- make sure Glargine is started

- follow up 8PM BMP

- monitor her respiratory status Q4h

- monitor fever cure

Situation & Contingency

- Replete her electrolytes if K <4, Phos <2, and Mg <2.

- If anion gap increases insulin drip may need to restart

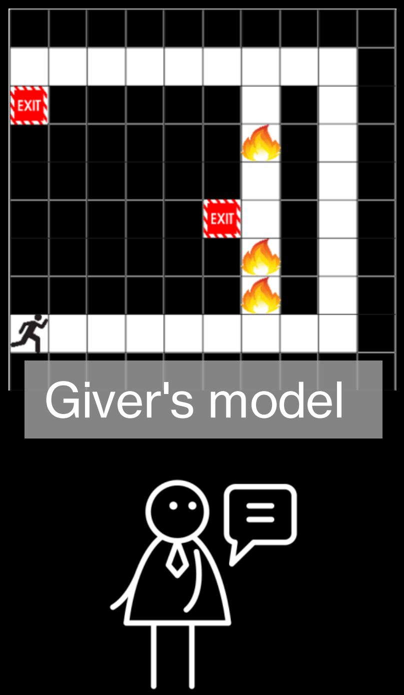
- For resp distress get a CXR and consider Lasix 40 mg IV

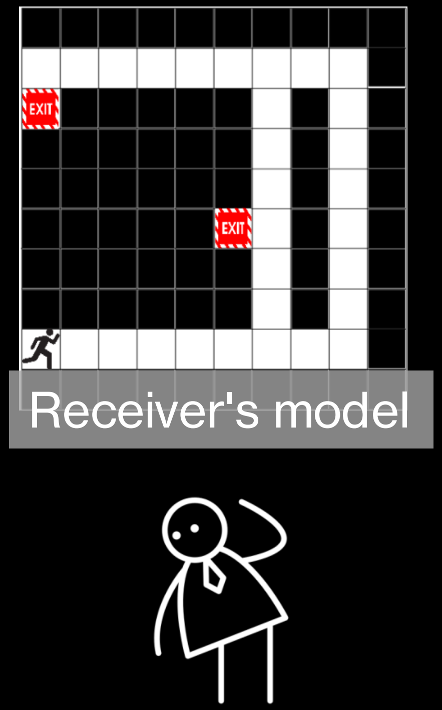
- If she develops a fever > 101F, please re-culture urine and blood and broaden antibiotics to Zosyn.

Synthesis by Receiver

An example synthesis will be included in the spoken handoff that follows.

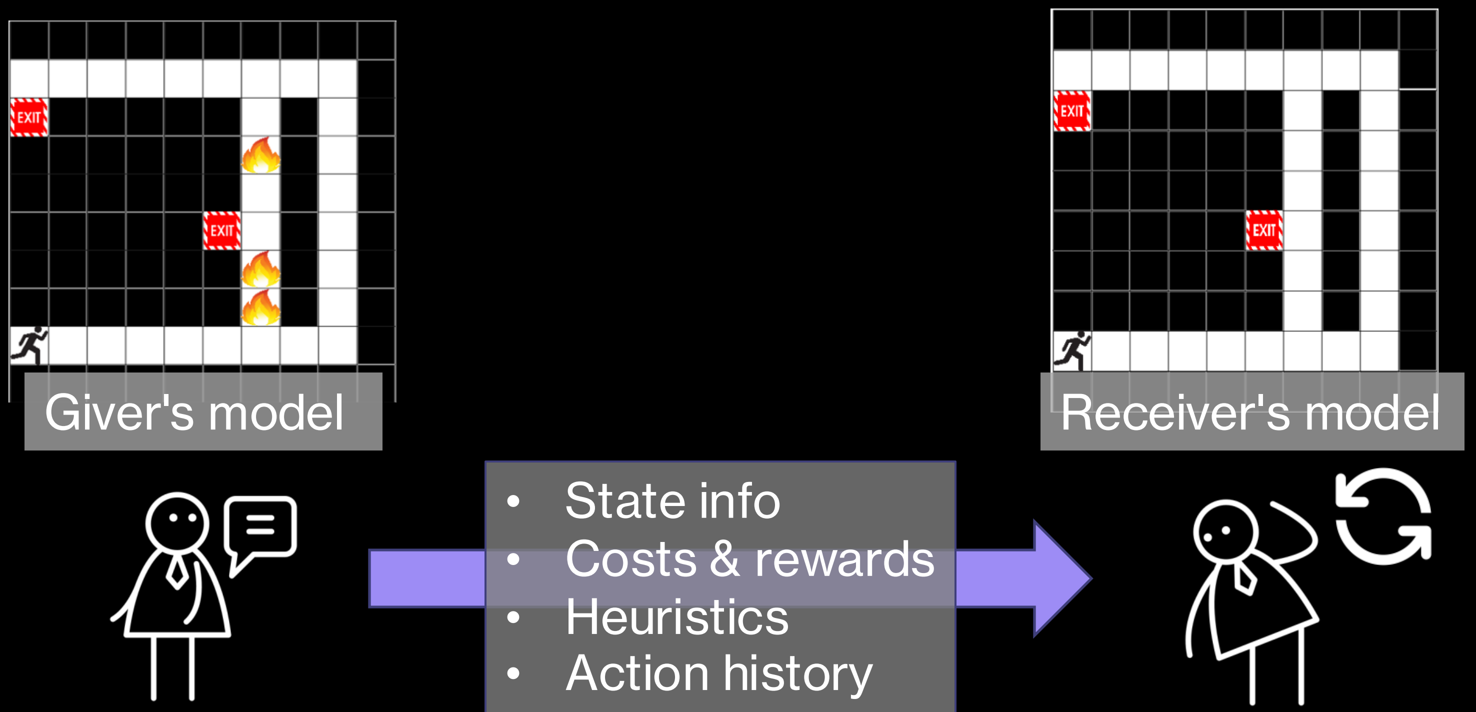
Task handover as model state reconciliation



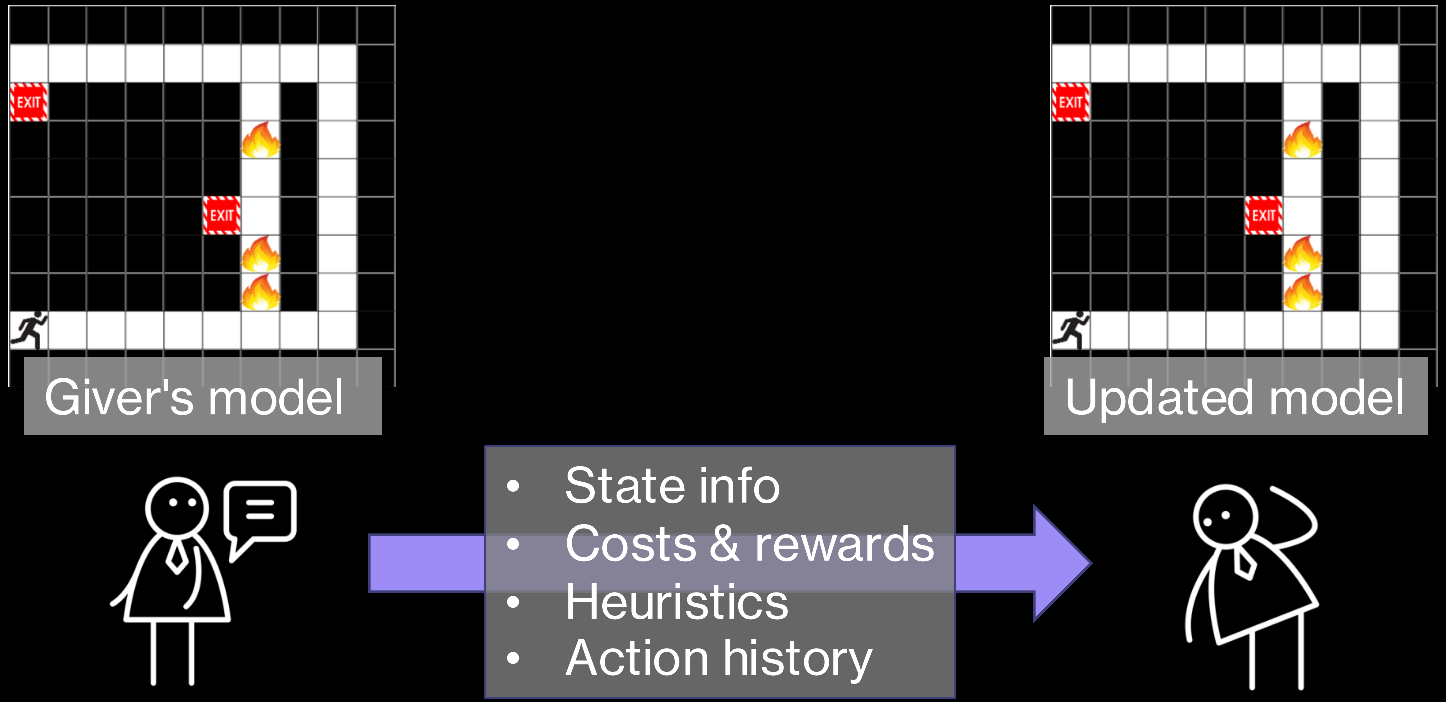


[Images: Tabrez et al., 2021]

Task handover as model state reconciliation

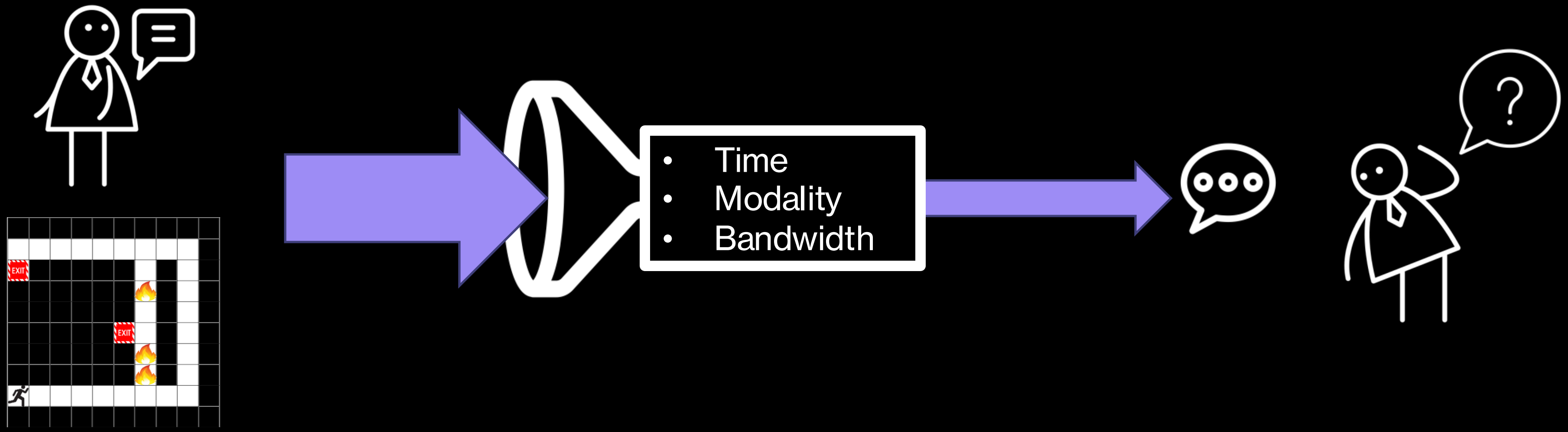


Task handover as model state reconciliation

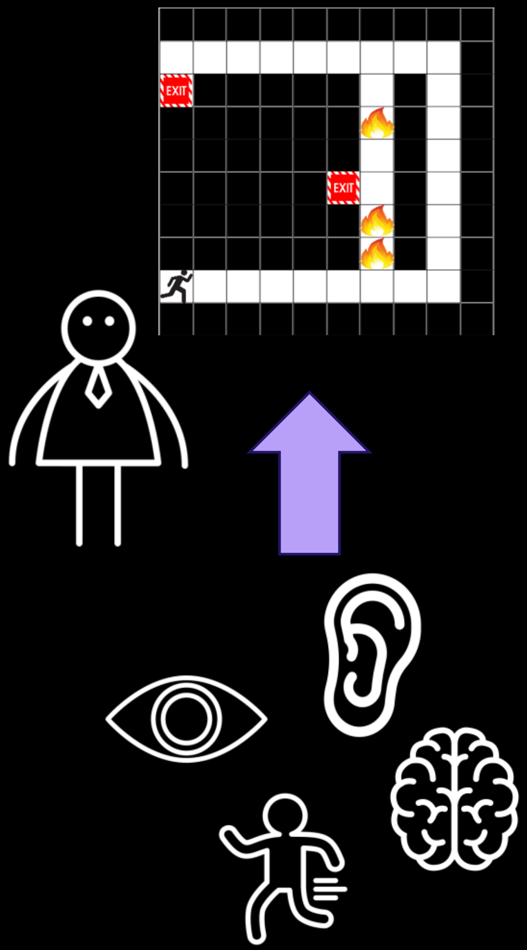


Task handover has constraints

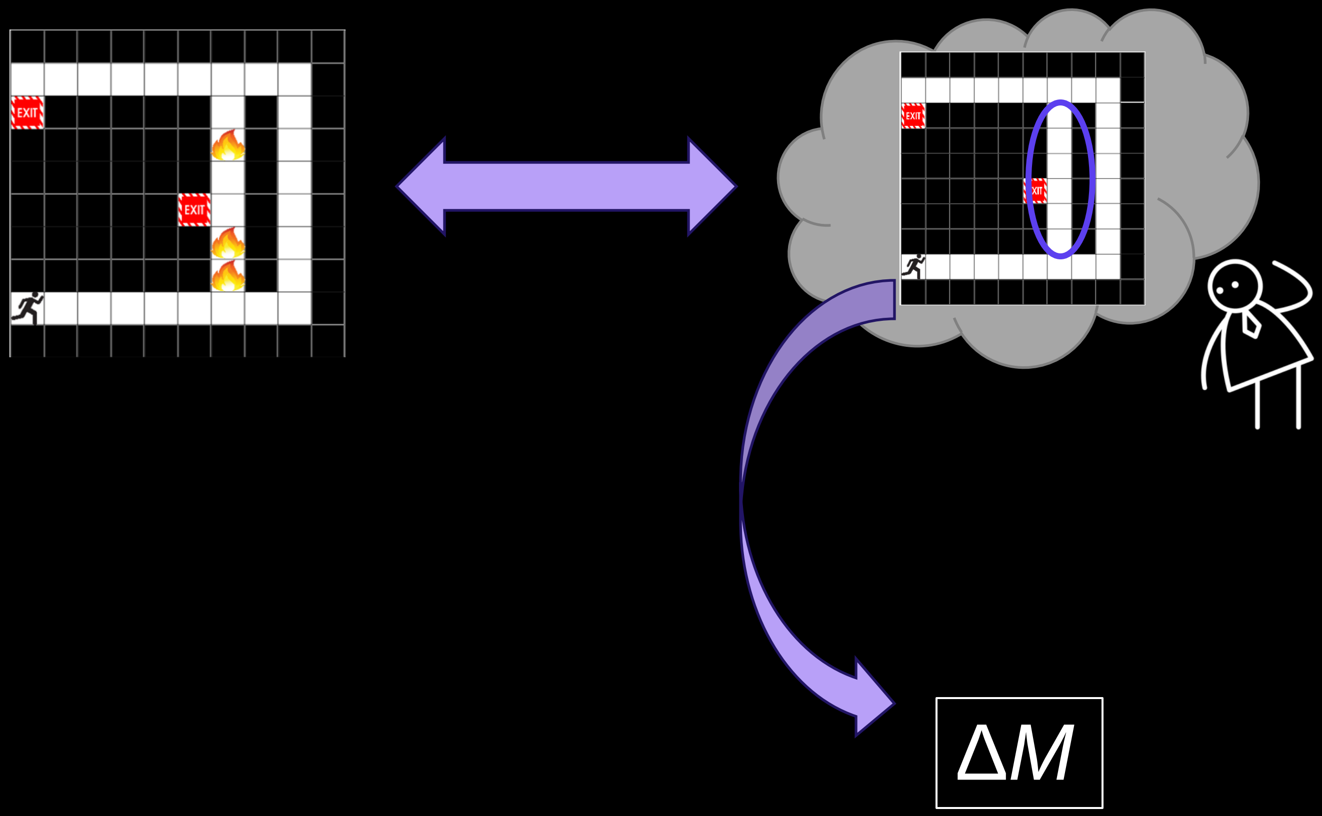
- Handover is usually verbal, with limited communication or prep time
- Humans can only retain so much information at once
- To address these constraints, handover has multiple stages



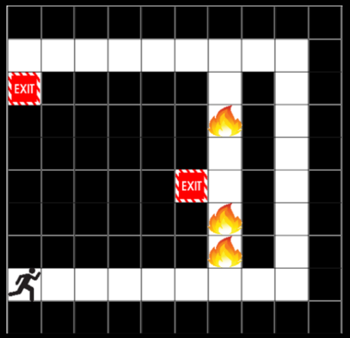
1. Form model state representation from input data

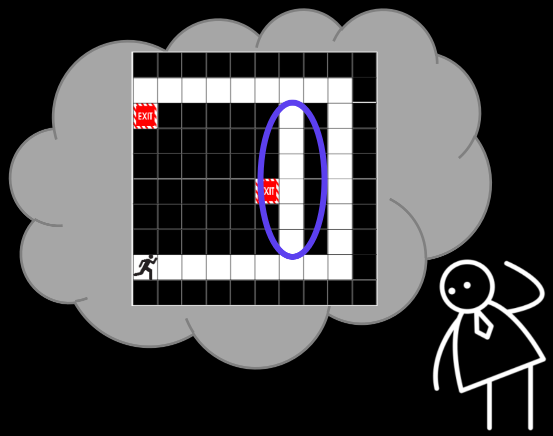


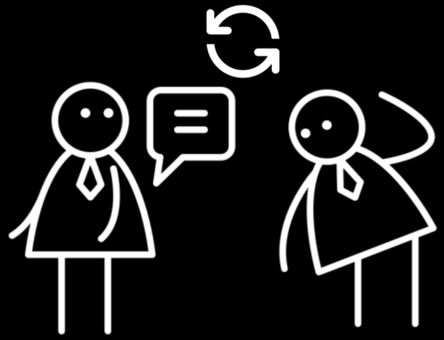
2. Estimate receiver model state (and model difference)

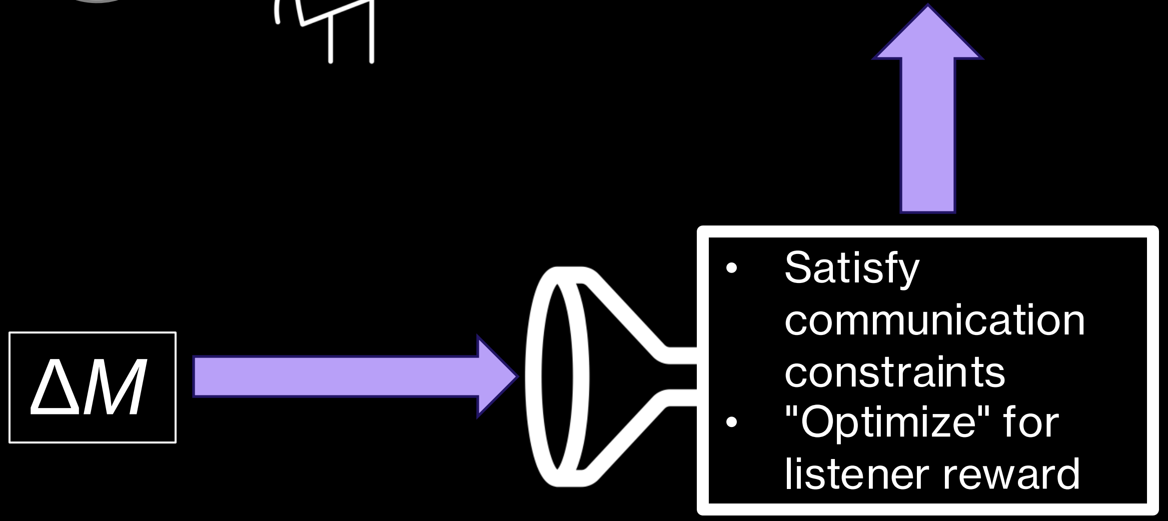


3. Generate handover communication for model update

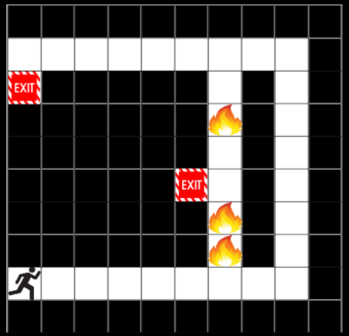




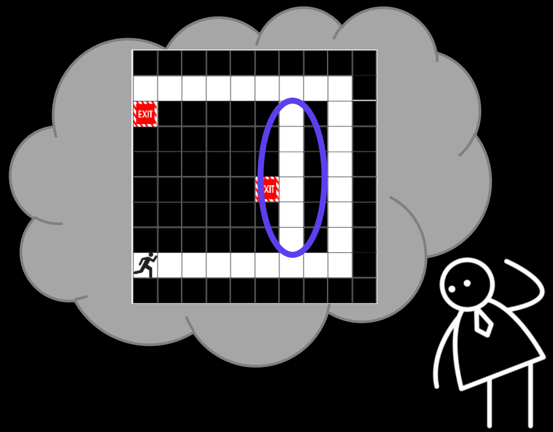




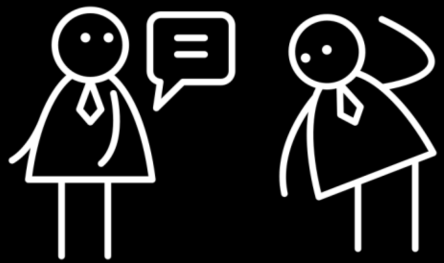
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)



3. Generate handover communication for model update



Task handover is difficult

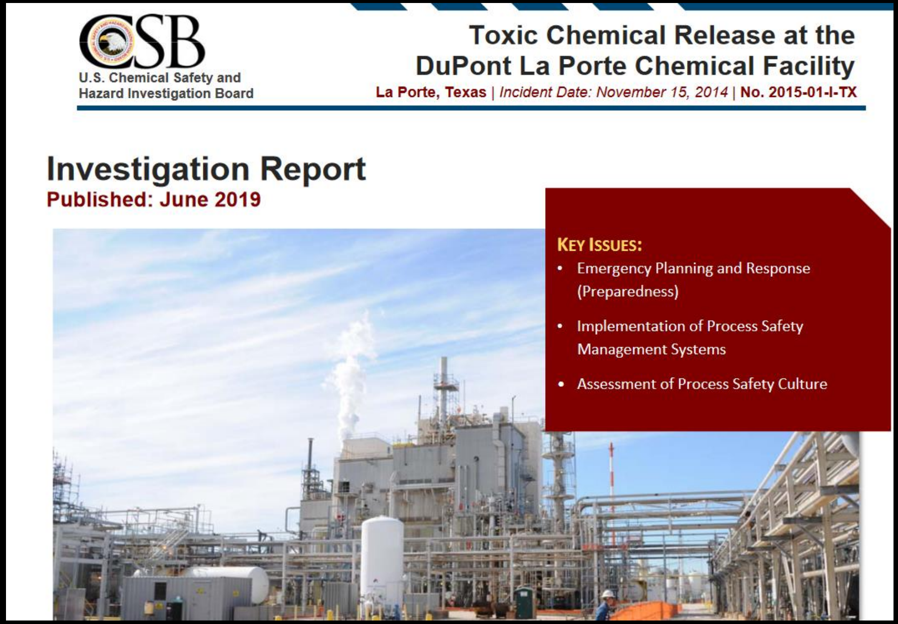
- …and people make mistakes – with severe consequences
  - 80% of preventable adverse medical events (Joint Commission 2017)
  - U.S. Chemical Safety & Hazard Investigation Board has attributed multiple fatal industrial accidents, in part, to failure to communicate key information at shift handover


Sentinel Event Alert

A complimentary publication of The Joint Commission
Issue 58, September 12, 2017

Published for Joint Commission-accredited

Inadequate hand-off communication


CSB
U.S. Chemical Safety and Hazard Investigation Board

Toxic Chemical Release at the DuPont La Porte Chemical Facility
La Porte, Texas | *Incident Date: November 15, 2014* | No. 2015-01-I-TX

Investigation Report
Published: June 2019

KEY ISSUES:
- Emergency Planning and Response (Preparedness)
- Implementation of Process Safety Management Systems
- Assessment of Process Safety Culture

Why should AI participate in task handover?

- The incorporation of AI assistance into human-human handover can reduce miscommunication errors, as well as temporal & cognitive burden
- AI-to-human handover may be necessary in human-robot collaboration scenarios
- Despite this, almost no work has characterized task handover as a computational problem of conveying a task and/or world state, and none has investigated applying AI

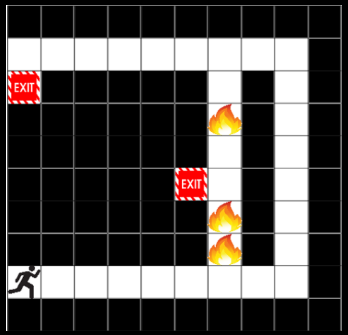
[Image: Roncone et al., 2017]

Why should AI participate in task handover?

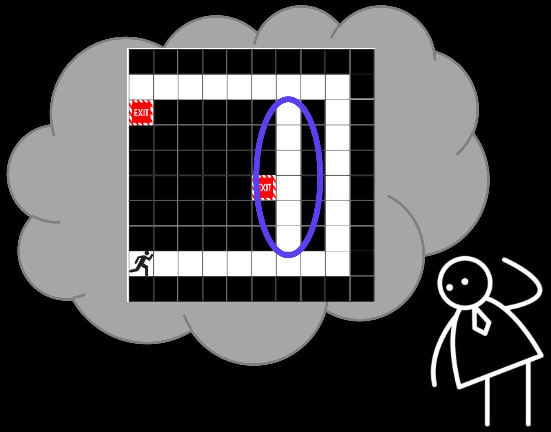
1. Handover is an important, yet underserved problem space
2. Recent advances in related areas of AI can be applied to task handover
3. LLMs have made these advances more applicable to real-world communication scenarios

Tackling the component parts of handover

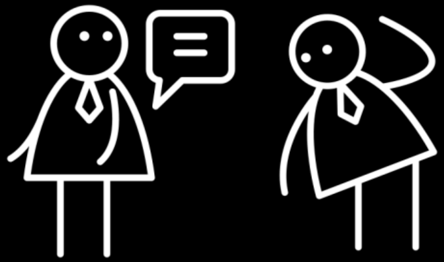
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)



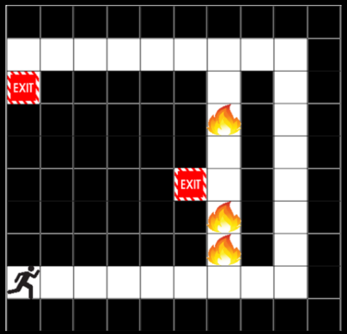
3. Generate handover communication for model update



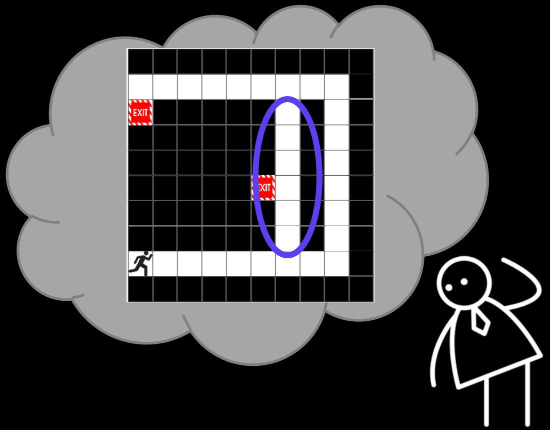
Tackling the component parts of handover

Literature review

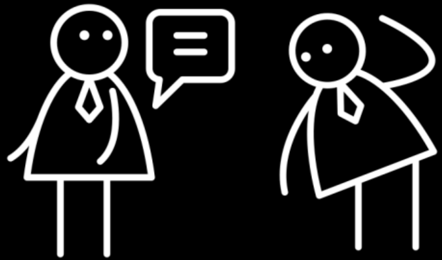
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)



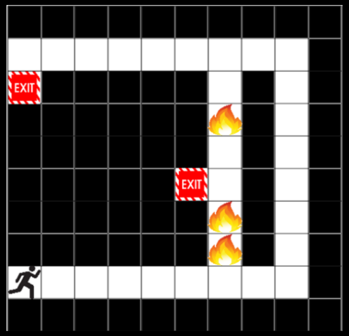
3. Generate handover communication for model update



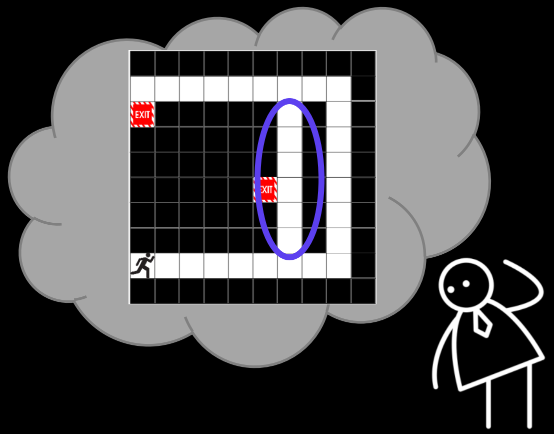
Tackling the component parts of handover

Current work & future directions

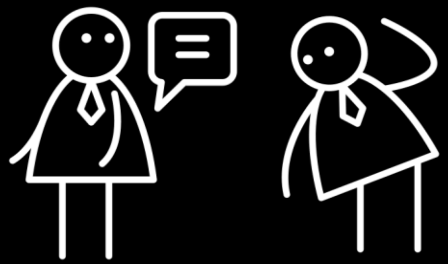
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)

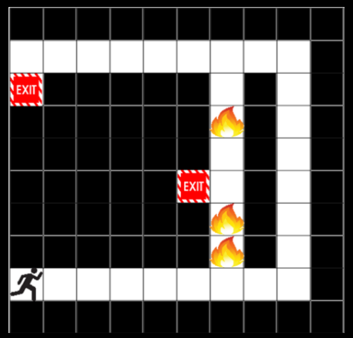


3. Generate handover communication for model update

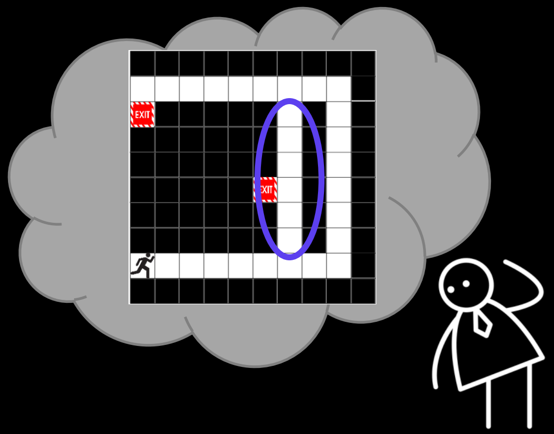


Tackling the component parts of handover

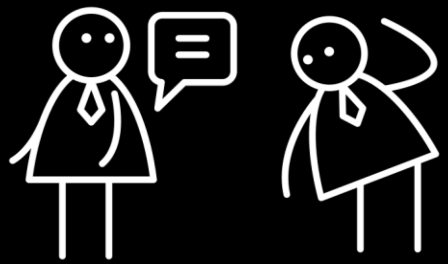
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)

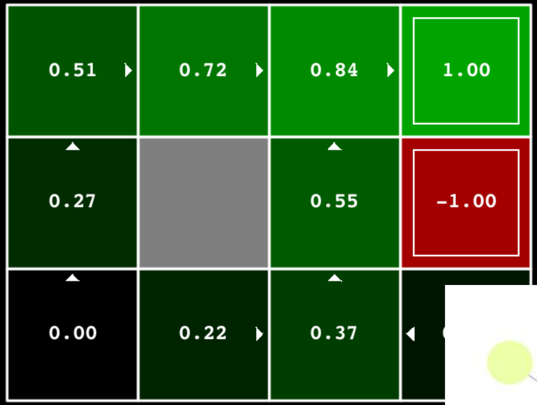


3. Generate handover communication for model update

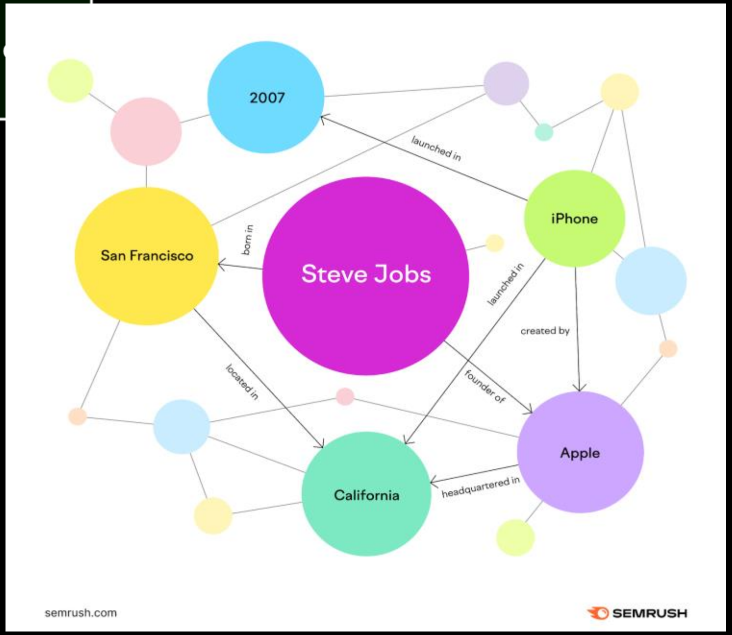


Primer: Forming state representations for handover

- For handover, state representations need to be:
  - Communicable via natural language
  - Compatible with a notion of goals and/or task-relevance
- Many real state spaces are functionally unbounded in nature
- Natural language-based techniques are an excellent tool for creating and utilizing these representations



VALUES AFTER 5 ITERATIONS



Hierarchical event representations from multimodal data

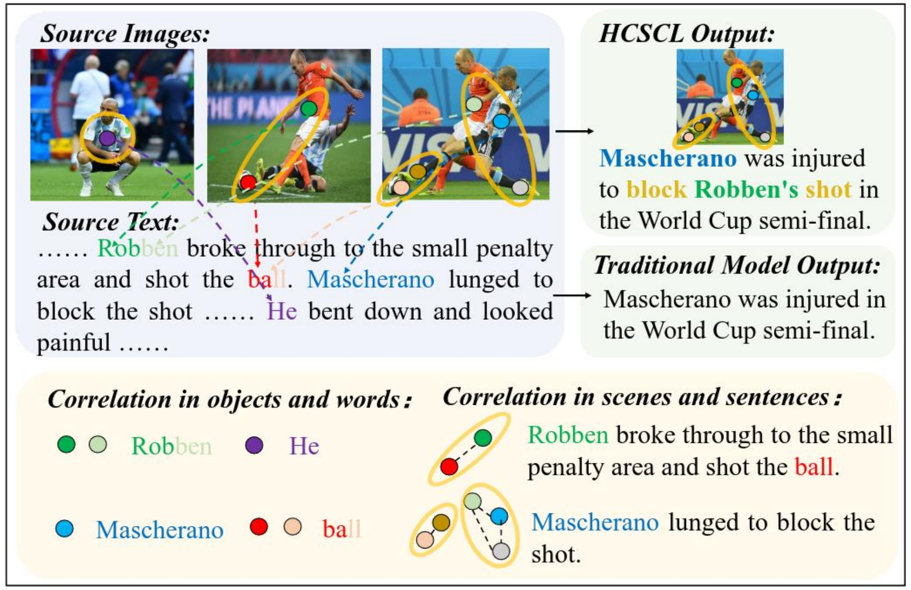
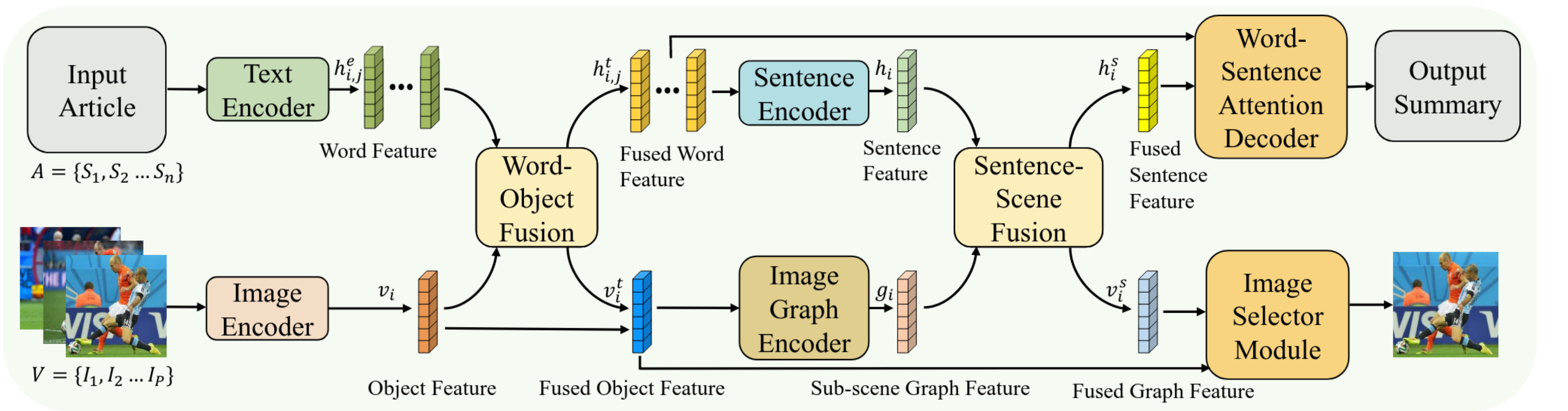


Figure 1: An example of the cross-modality semantics correlation in multimodal data.

- Multimodal (text + images) summarization from news articles
- Using a hierarchical grounding approach of both word-object and sentence-scene grounding,
- Connecting, combining and condensing data from text and image input into a semantic representation

Zhang, L., Zhang, X., & Pan, J. (2022). Hierarchical Cross-Modality Semantic Correlation Learning Model for Multimodal Summarization. Proceedings of the AAAI Conference on Artificial Intelligence, 36(10), 11676–11684.

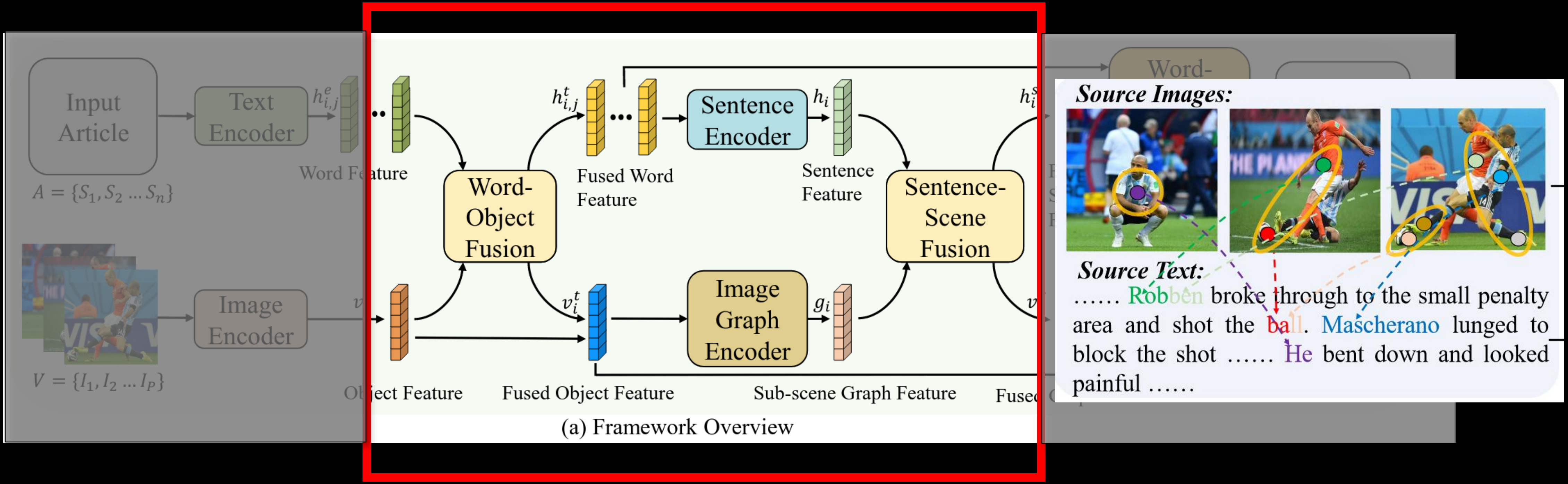
Hierarchical event representations from multimodal data



(a) Framework Overview

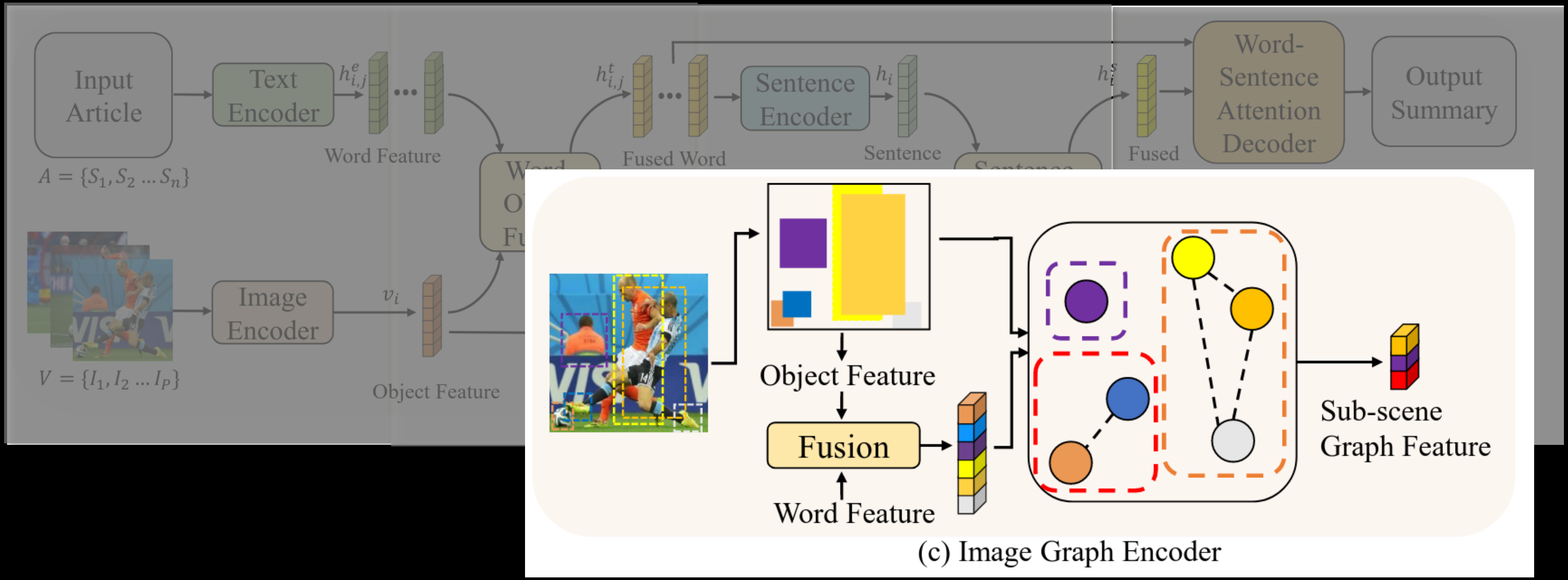
Zhang, Zhang, & Pan, 2022

Hierarchical event representations from multimodal data



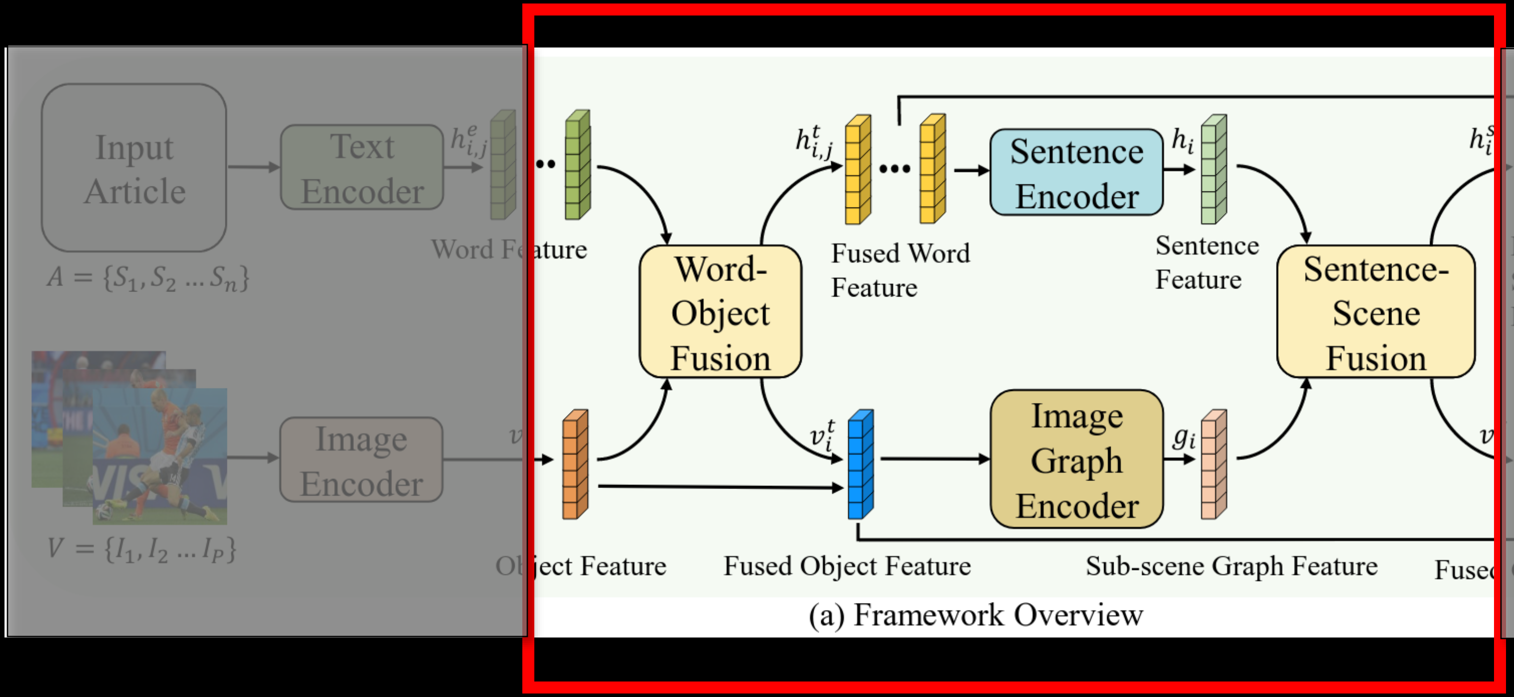
Zhang, Zhang, & Pan, 2022

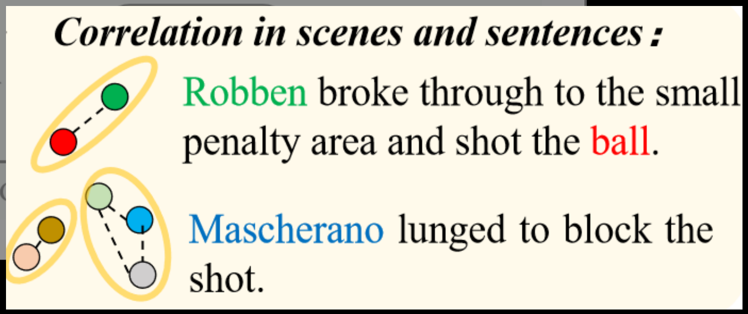
Hierarchical event representations from multimodal data



Zhang, Zhang, & Pan, 2022

Hierarchical event representations from multimodal data

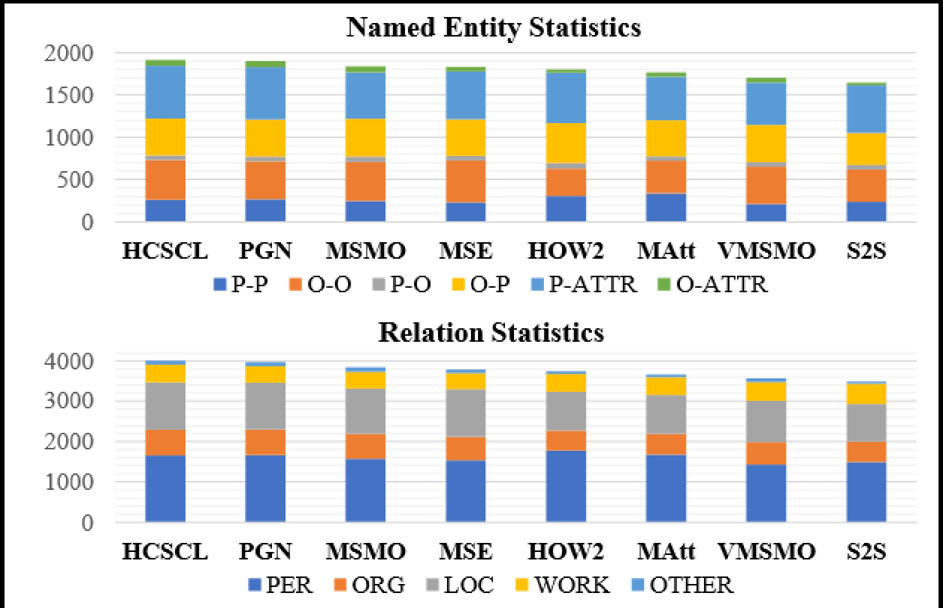




Hierarchical event representations from multimodal data

- Outperforms baselines (multimodal and text-only) by around 1% on average across standard summarization metrics

- Summaries have greater numbers of named entities and relations



Zhang, Zhang, & Pan, 2022

Hierarchical event representations from multimodal data

Key point: Can develop grounded concept and relationship representations by fusing info across multiple modalities

With LLMs/MLMs, we might expect see much higher generalizability across domains, and an ability to condition on specific task information / relationships

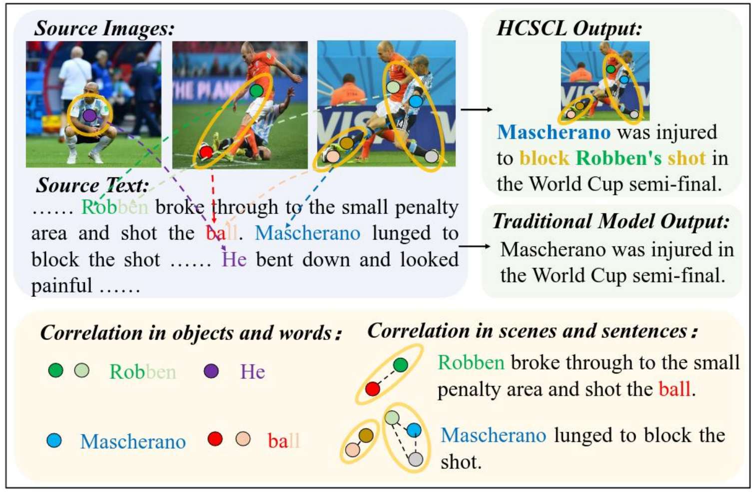
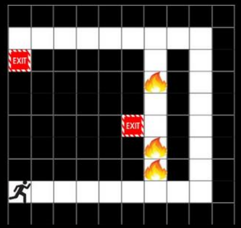
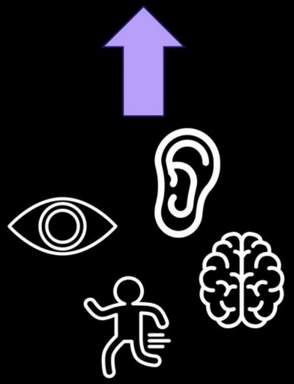


Figure 1: An example of the cross-modality semantics correlation in multimodal data.

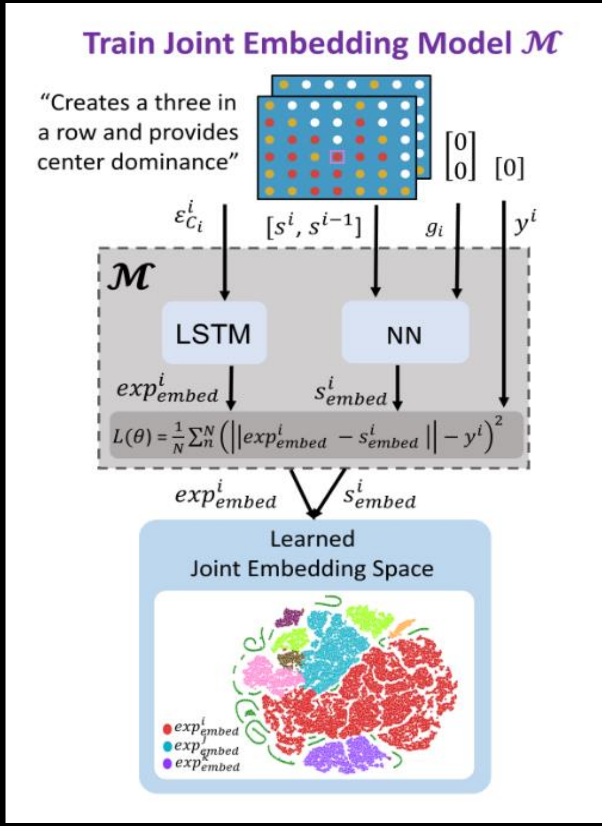




Zhang, Zhang, & Pan, 2022

State2Explanation: Building concept-grounded internal models

- Agent has a task model of its own, including a RL-trained policy
- Learns a joint embedding model between agent state-action pairs and associated high-level concepts
- Use model to generate explanations for state-action pairs for end user

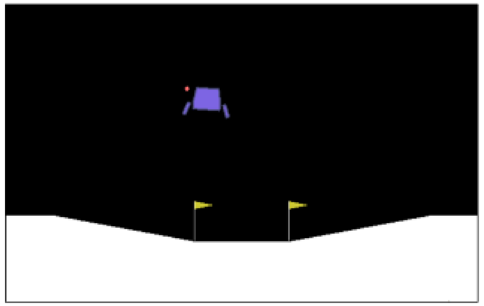


Das, D., Chernova, S.., & Kim, B. (2023). State2Explanation: Concept-Based Explanations to Benefit Agent Learning and User Understanding. Advances in Neural Information Processing Systems, 36, 67156–67182.

State2Explanation: Building concept-grounded internal models

- "concepts" = grounded in human domain knowledge, goal-related, and generalizable
- Multiple states can be associated to multiple, non-unique concepts

Lunar Lander Concepts	Description
POS	Brings lander closer to center
VEL	Decrease lander speed to avoid crashing
TILT	Decrease tilt
RLEG	Encourage right leg contact with ground
LLEG	Encourage left leg contact with ground
MF	Conserve main fuel usage
SF	Conserve side fuel usage
L	Land without crashing



$\varepsilon_{concept}$: Fire left engine because it moves lander closer to the center, decreases lander speed to avoid crashing, decreases tilt of lander, and conserves main fuel usage.

Das, Chernova, & Kim, 2023.

State2Explanation: Building concept-grounded internal models

- Learn joint embedding model to align (*s, a)* pair with a set of concept embeddings

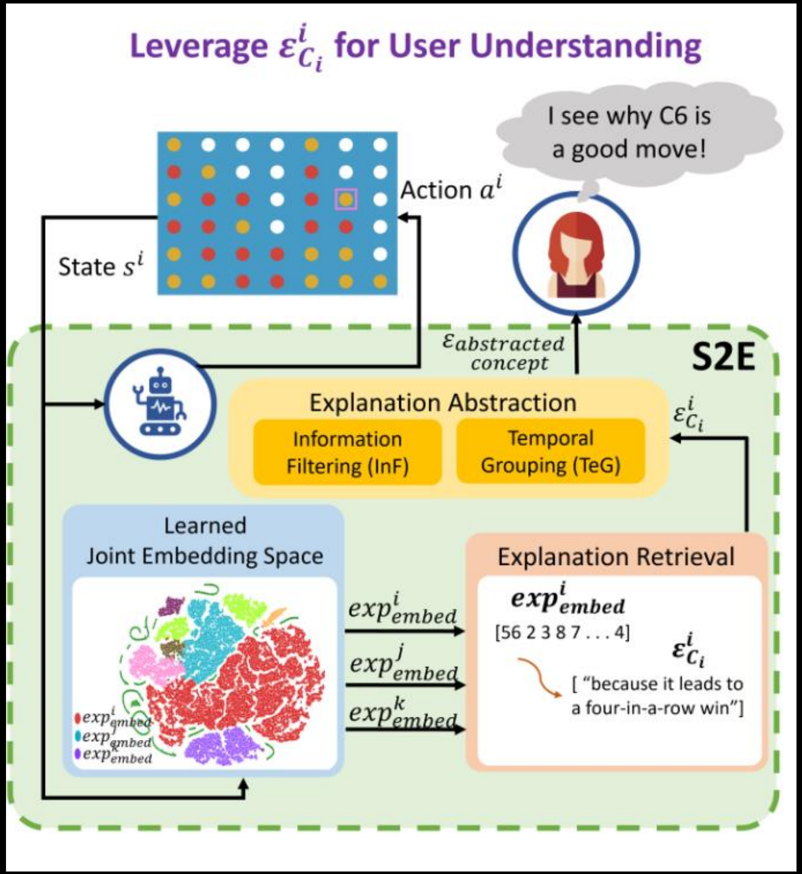
- Only return concepts that immediately influence ability to reach goal

"Fire left engine because it brings lander closer to the center, decreases lander velocity to avoid crashing, and decreases tilt"

becomes simply

"...because it brings lander closer to the center"

if velocity and tilt (as concepts) do not currently impact success probability much



Das, Chernova, & Kim, 2023.

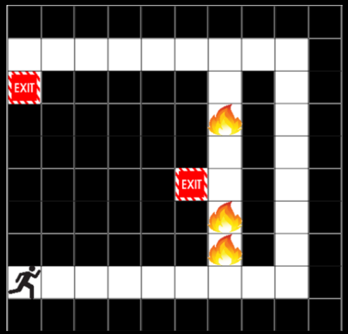
State2Explanation: Building concept-grounded internal models

- **Key point:** When given a task model, policy, and domain knowledge, agents can create an explanation-grounded internal model that can then be used to **generate** & **filter** relevant explanations, and could (ideally) generalize to new states with the same concepts

- With LLMs, the generation of these explanations can become more flexible than template-based approaches allow
  - Concepts can be drawn from LLM knowledge, rather than hand-crafted
  - Dialogue can be used to prompt for clarification, such as asking for more detail or counter-examples

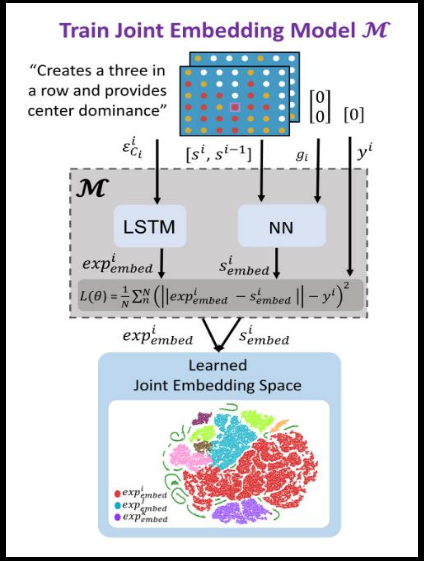
Recap: Forming state representations for handover

1. Form model state representation from input data



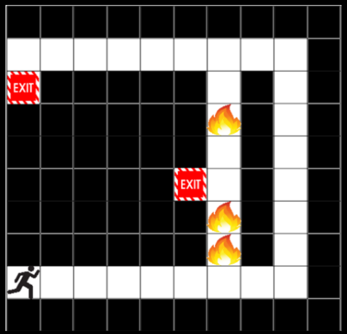
1. Extracting condensed semantic representations of events from text & image data (that retains key entities and relationships)

2. Mapping task states to higher-level semantic concepts, and relating those concepts to achieving a goal

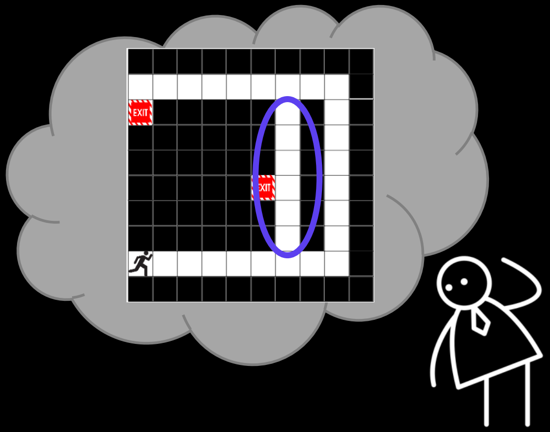


Tackling the component parts of handover

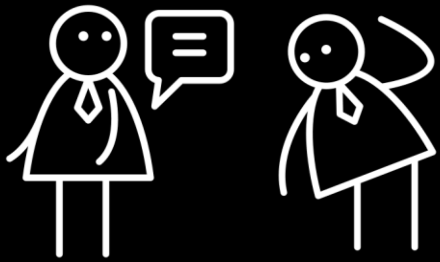
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)

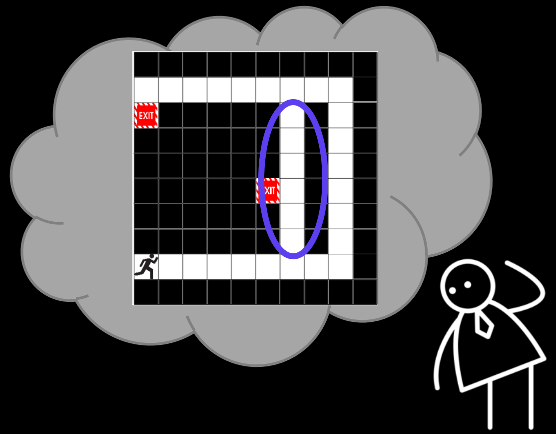


3. Generate handover communication for model update



Tackling the component parts of handover

2. Estimate receiver model state (and model difference)



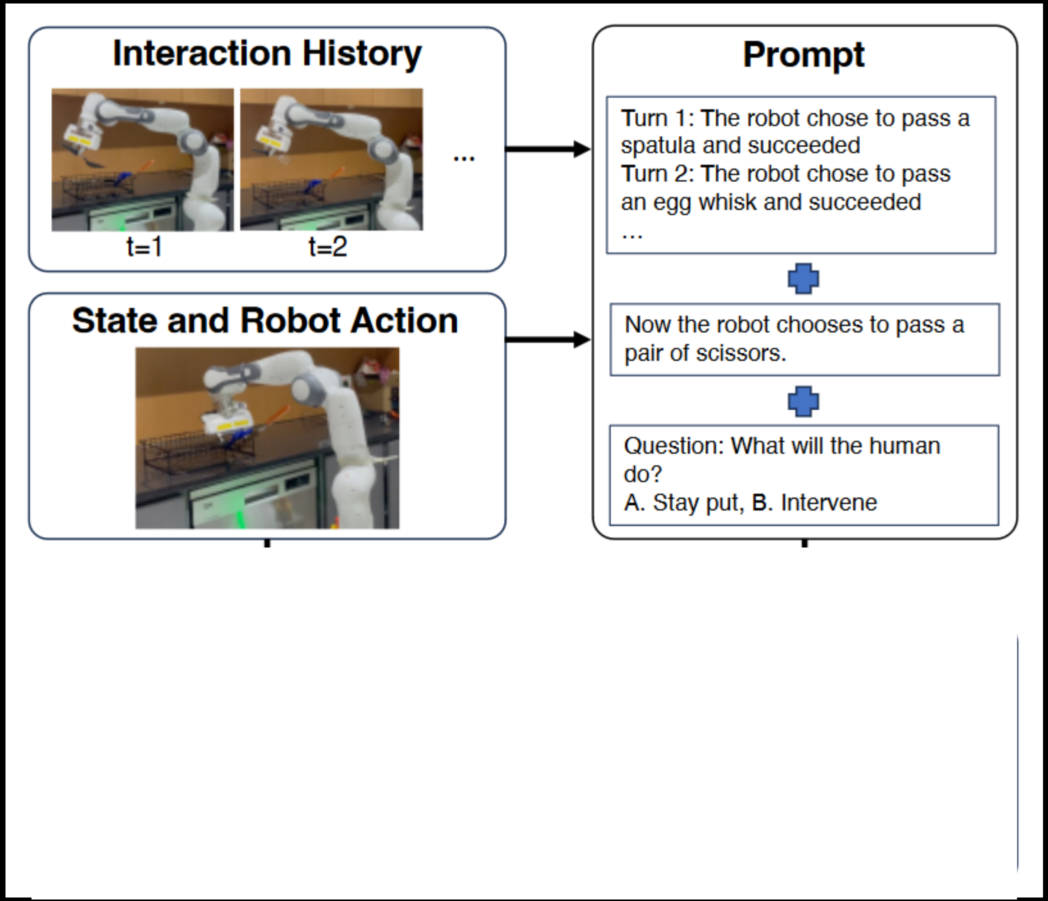
A. LLMs as no-shot human models

B. Knowledge tracing
   - Interactively
   - With deep learning

C. Social projection for model state inference

LLM as a no-shot human model

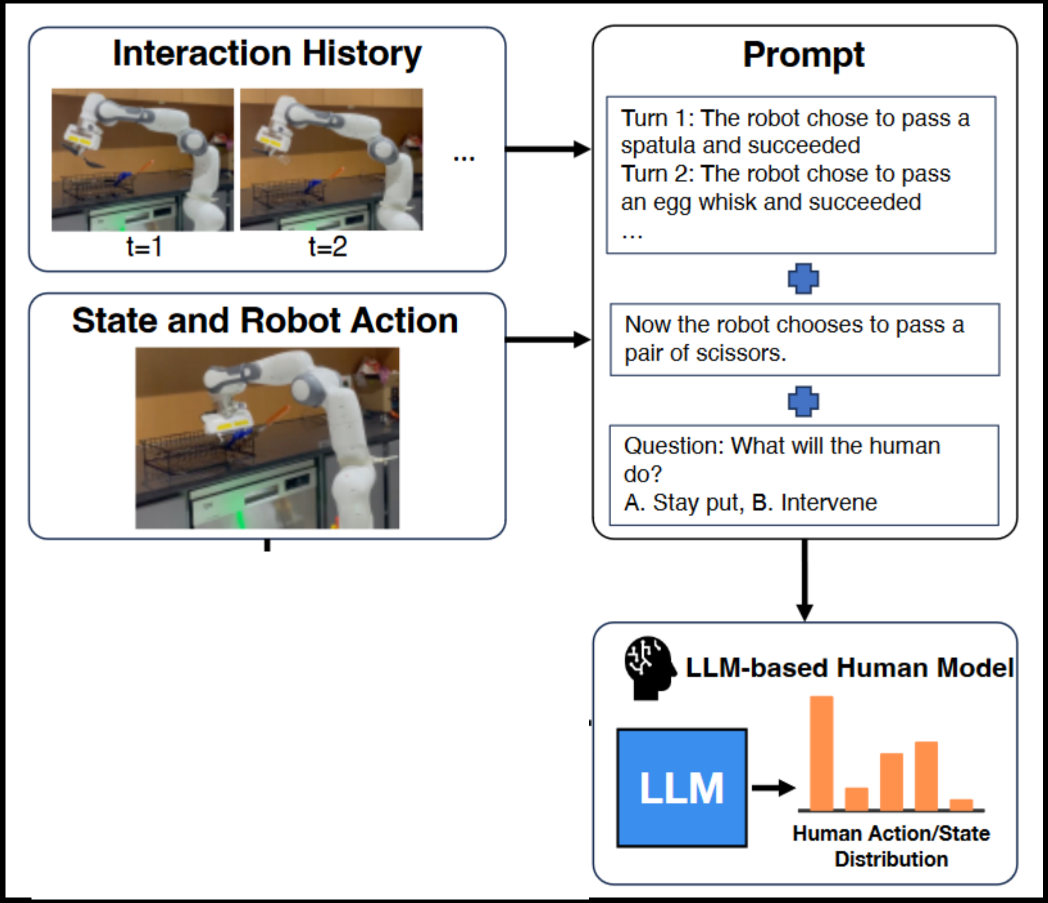
- With LLMs, in some applications we might not need a bespoke human model if we can get one for free
- Zhang & Soh explore use of LLMs as human models for predicting trust dynamics and human actions in HRI planning problems



Zhang, B., & Soh, H. (2023). Large Language Models as Zero-Shot Human Models for Human-Robot Interaction. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 7961–7968.

LLM as a no-shot human model

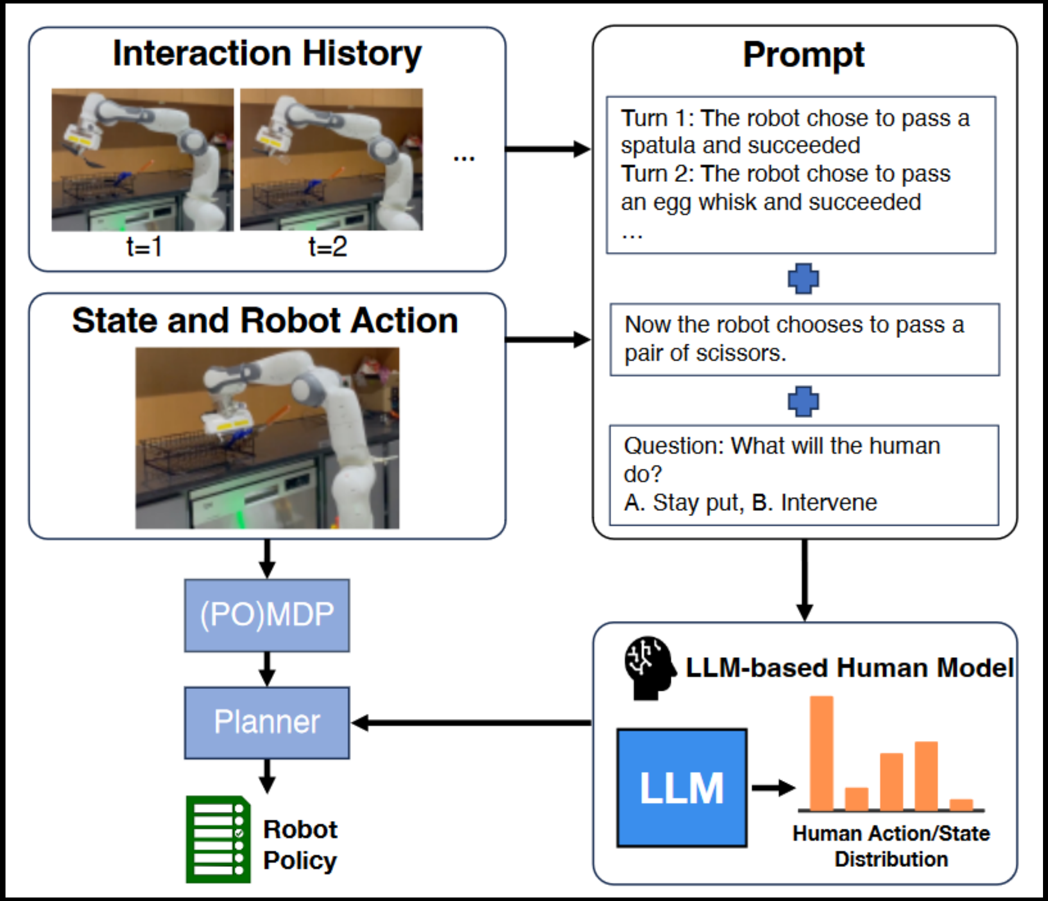
- With LLMs, in some applications we might not need a bespoke human model if we can get one for free
- Zhang & Soh explore use of LLMs as human models for predicting trust dynamics and human actions in HRI planning problems



Zhang, B., & Soh, H. (2023). Large Language Models as Zero-Shot Human Models for Human-Robot Interaction. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 7961–7968.

LLM as a no-shot human model

- With LLMs, in some applications we might not need a bespoke human model if we can get one for free
- Zhang & Soh explore use of LLMs as human models for predicting trust dynamics and human actions in HRI planning problems



Zhang, B., & Soh, H. (2023). Large Language Models as Zero-Shot Human Models for Human-Robot Interaction. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 7961–7968.

LLM as a no-shot human model

- **Key point:** LLMs have a lot of "common sense" reasoning power about human behavior
- Can help to make reasonable assumptions about a human's world knowledge and/or policy
- This was on an older model (OpenAI's DaVinci) that still struggled on numerical/spatial tasks – can expect better performance from newer models

Estimating human states with knowledge tracing

Knowledge tracing: modeling the knowledge state of the user

- Frequently used in intelligent tutoring systems
- Inferring knowledge from observations of behavior
- Many approaches, from probabilistic to deep learning

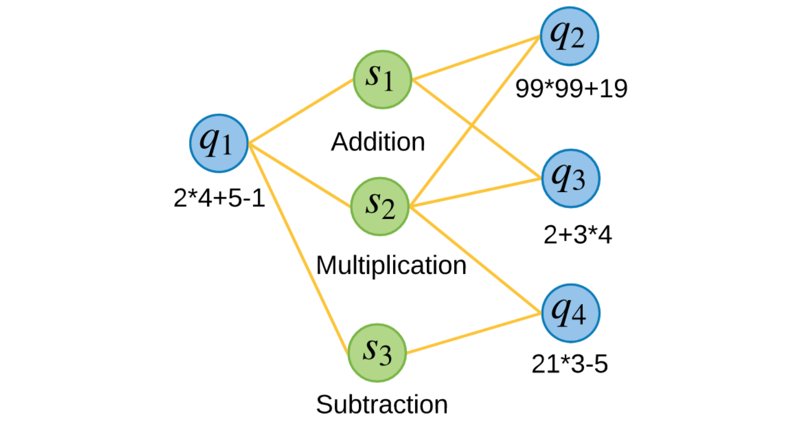
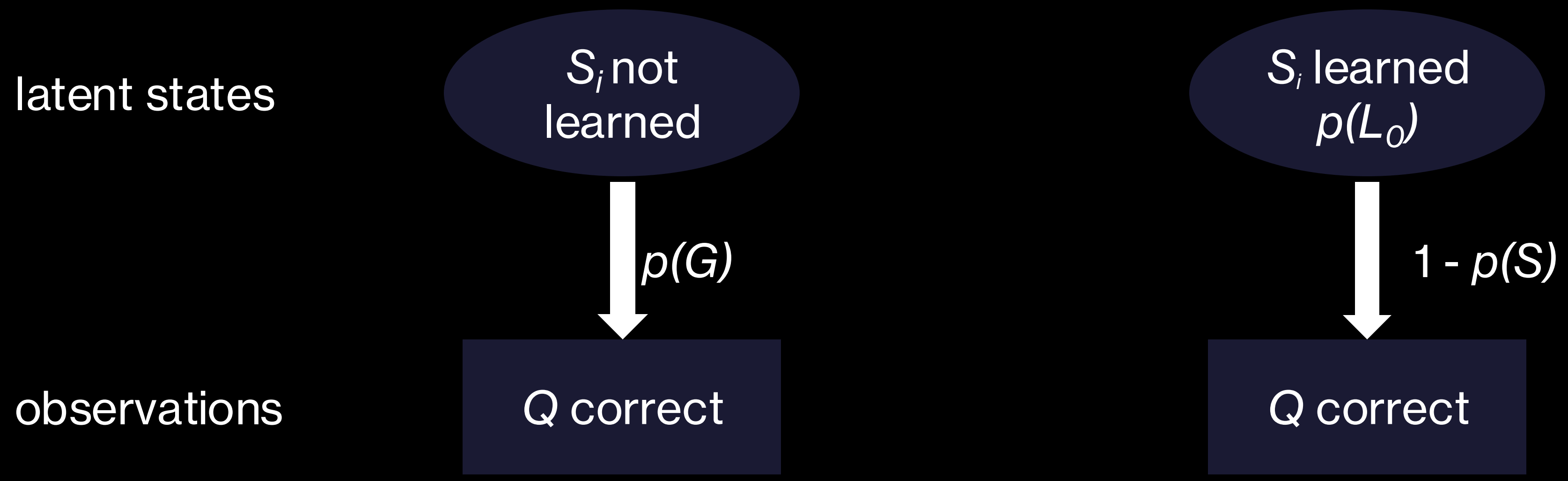


Fig. 1. Simple example of question-skill relation graph.

Primer: Bayesian knowledge tracing

- A single observation might not be enough to infer knowledge
- Bayesian knowledge tracing (BKT) models the state (and learning dynamics) of the user from binary correct/incorrect question performance



p(G): probability that the user just guessed correctly

p(S): probability that the user slipped (but knows the skill)

Knowledge tracing using POMDPs

- What if we could *query* the knowledge state of the user?
- Using a POMDP for this would ordinarily be intractable

- Insight: apply **BKT** dynamics to POMDP observation model
- Reward based on *information gain* – how much the estimate of the user's skill was updated by their success/failure at that task

Salomons, N., Akdere, E., & Scassellati, B. (2021). BKT-POMDP: Fast Action Selection for User Skill Modelling over Tasks with Multiple Skills. Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, 4243–4249.

Knowledge tracing using POMDPs

POMDP:

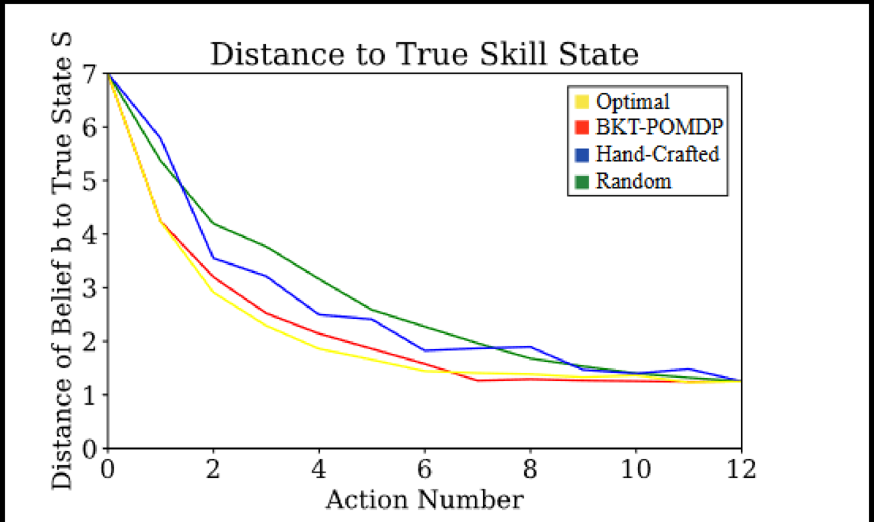
- **S** : true skill state of the user, hidden from the agent <0, 1, 1, 0.... >
- **b** : skill belief vector, agent's current estimate of *S*
- **A** : action space; tasks that can be presented to the user, each associated with 1+ skills
- **O** : observation probabilities *P(o|b, a)* from Bayesian Knowledge Tracing..
- **T** : transition function updating belief, *b' = P(b|o)*
- **R** : reward function. Here, reward is based on information gain of belief updates (KL(*b, b'*))
- **Ω** : observation space (0 for incorrect, 1 for correct, 2 for not tested, for each skill)

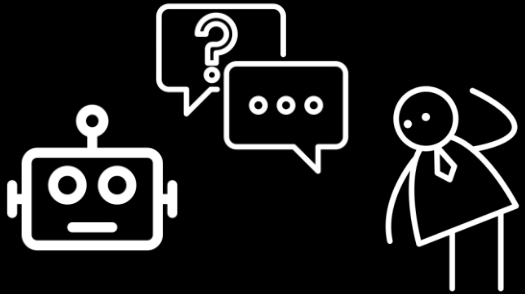
Salomons, Akdere, & Scassellati, 2021.

$$Q^*(b,a) = \sum_{o \in O} [P(o|b,a) \cdot R(b,b')]$$

Knowledge tracing using POMDPs

- **Key point:** Agent can query user's state to reduce uncertainty, without relying on single observations
- Using LLMs, the agent's action space can also take the form of **natural language,** and skill-action associations can be reasoned over dynamically





Salomons, Akdere, & Scassellati, 2021.

Knowledge tracing using deep learning

- What if our goal is just predicting behavior, but question-to-skill relationship is more nuanced?
- Graph convolutional network creates higher-order relation embeddings for questions & skills
- This embedding allows more nuanced mapping of questions to related skills that were not explicitly connected

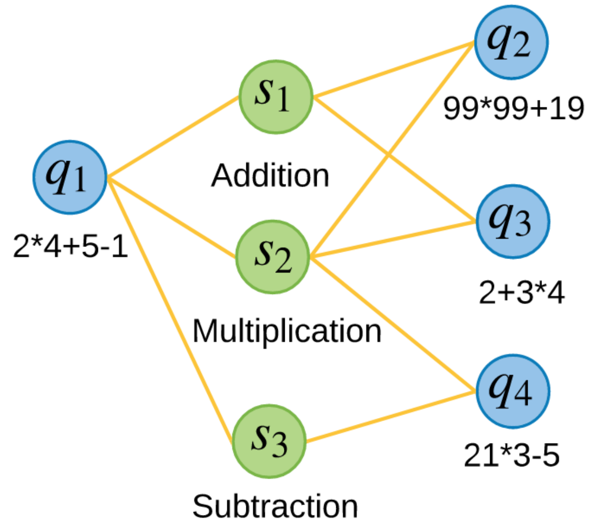
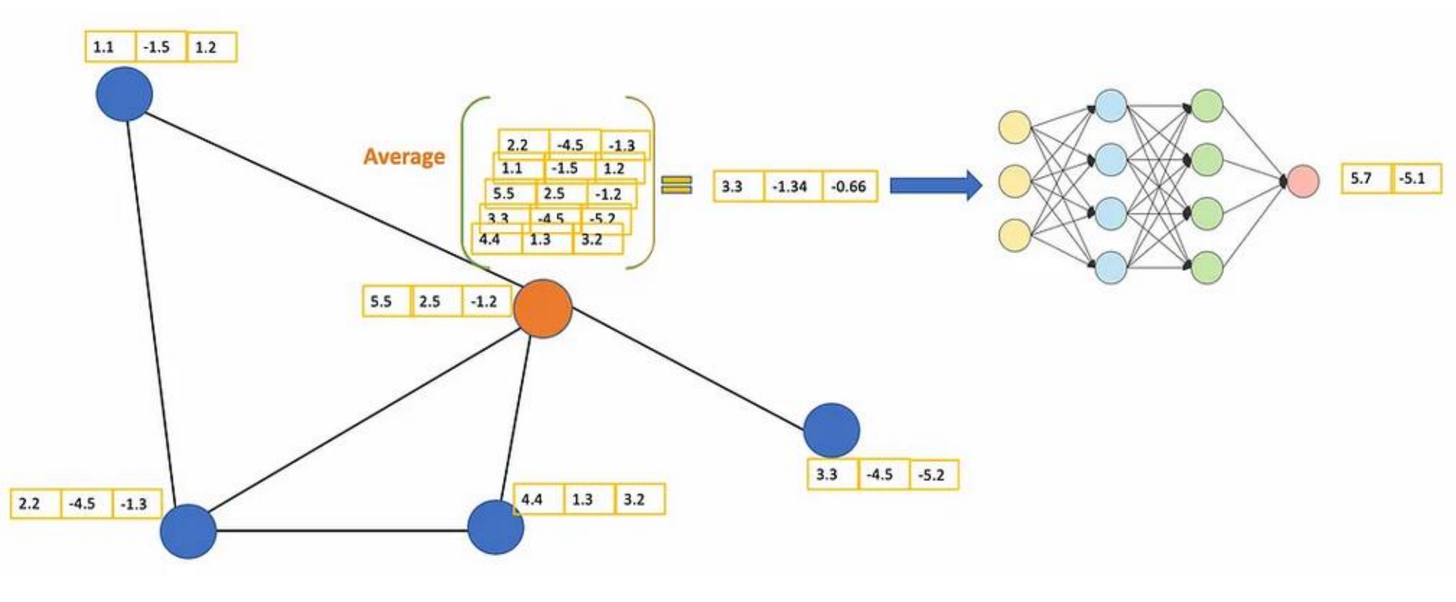


Fig. 1. Simple example of question-skill relation graph.

Yang, Y., Shen, J., Qu, Y., Liu, Y., Wang, K., Zhu, Y., Zhang, W., & Yu, Y. (2021). GIKT: A Graph-Based Interaction Model for Knowledge Tracing. In F. Hutter, K. Kersting, J. Lijffijt, & I. Valera (Eds.), Machine Learning and Knowledge Discovery in Databases (Vol. 12457, pp. 299–315). Springer International Publishing.

Sidebar: Graph convolutional networks

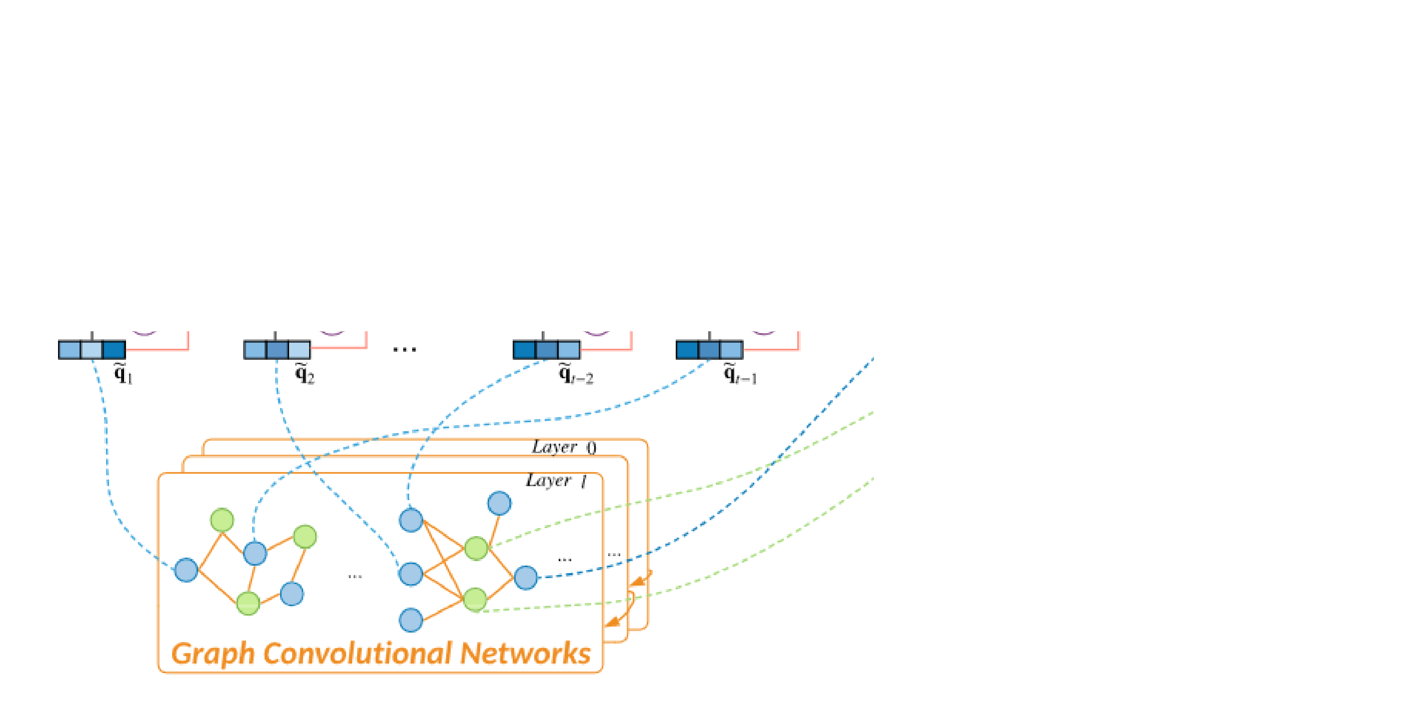


[Image: AI in Plain English]

- Input = a graph instance
- Each node feature vector is averaged together with that of its neighbors
- This can be done multiple times over the whole graph (embedding propagation)
- Result is passed through an MLP to get result embedding

Knowledge tracing using deep learning

- Goal: predict performance on question q_t
- GCN to capture nuanced question-skill relationships

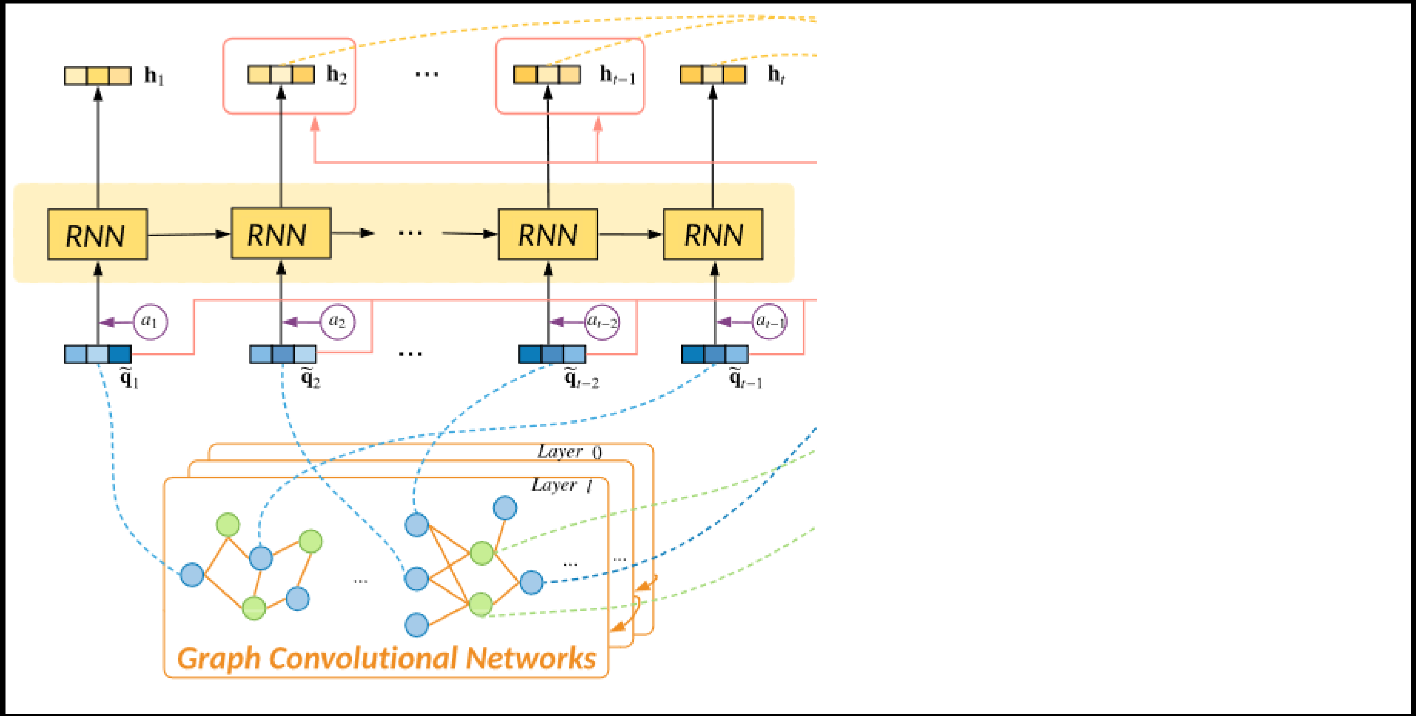


Yang et al., 2021

Knowledge tracing using deep learning

- Goal: predict performance on question q_t

- GCN to capture nuanced question-skill relationships
- LSTM to capture how past performance factors into current performance

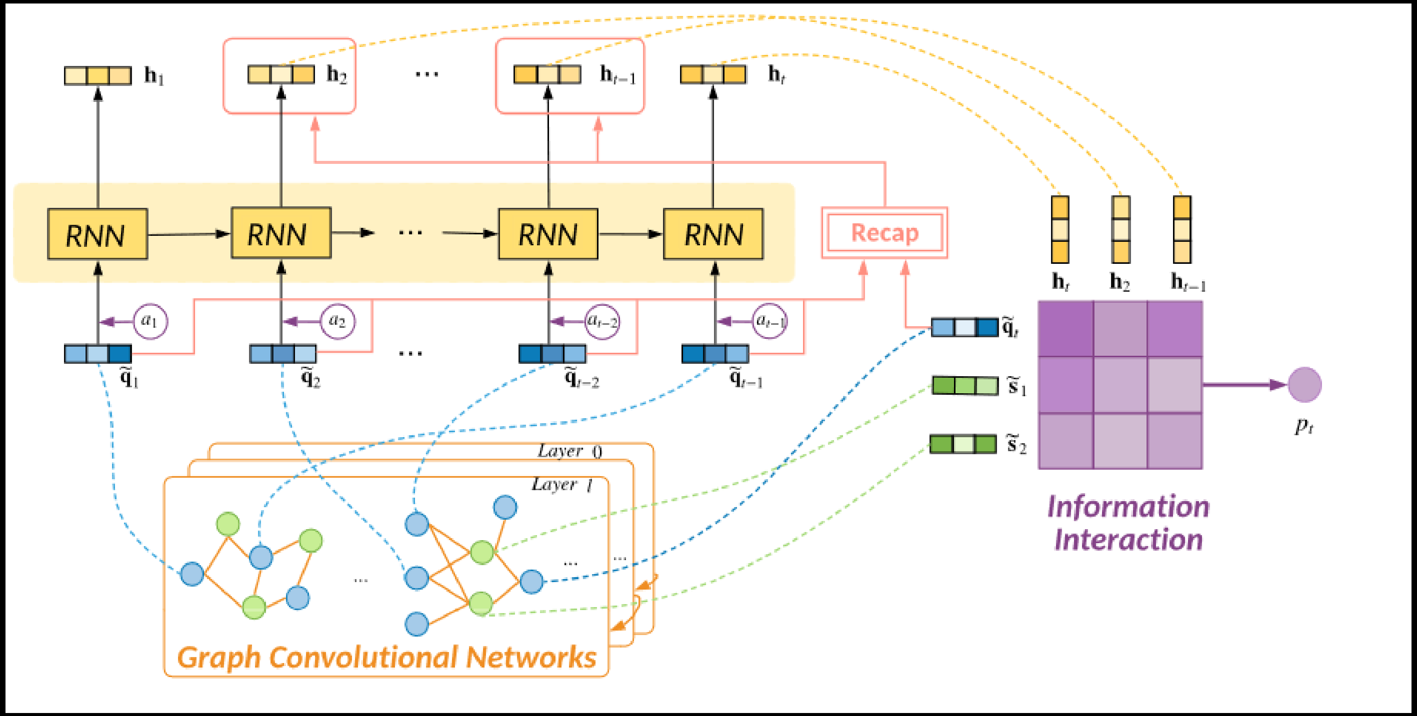


Yang et al., 2021

Knowledge tracing using deep learning

- Goal: predict performance on question q_t

- GCN to capture nuanced question-skill relationships
- LSTM to capture how past performance factors into current performance
- Interaction network to predict performance based on these embeddings



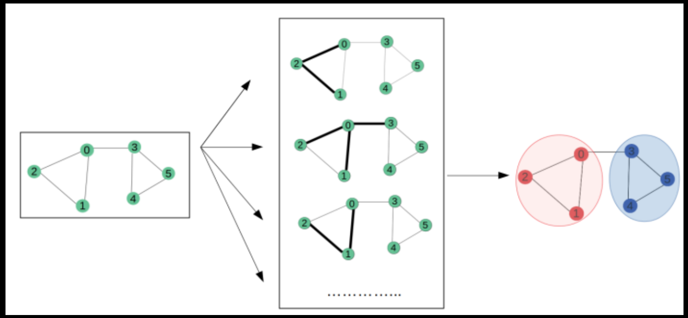
Yang et al., 2021

Knowledge tracing using deep learning

- **Key point:** Questions/observations and underlying skills can have more complex relationships that can be leveraged for prediction
- Skills can be grouped together if they share predictive qualities, giving rise to more high-level concept groupings
- By combining this with LLMs' ability to reason over semantic relationships between concepts, we can make better predictions about the user's knowledge & behavior, as well as generalize to new questions

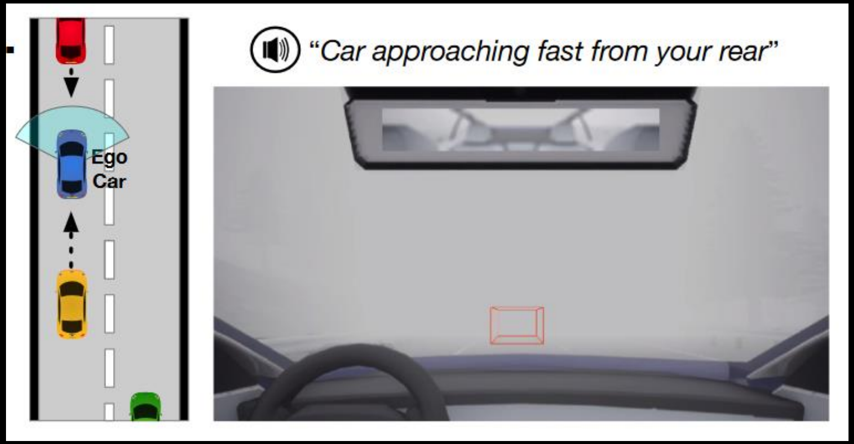
Yang et al., 2021

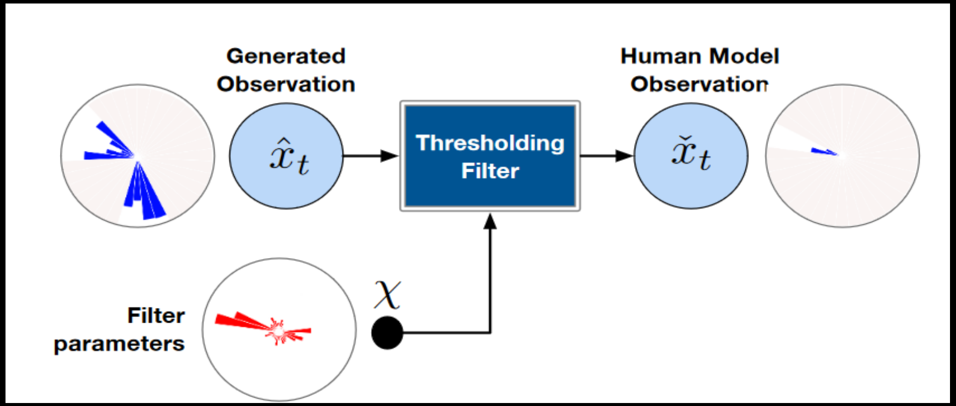
Model	ASSIST09	ASSIST12	EdNet
BKT	0.6571	0.6204	0.6027
KTM	0.7169	0.6788	0.6888
DKVMN	0.7550	0.7283	0.6967
DKT	0.7561	0.7286	0.6822
DKT-Q	0.7328	0.7621	0.7285
DKT-QS	0.7715	0.7582	0.7428
GAKT	0.7684	0.7652	0.7281
GIKT	**0.7996***	**0.8278***	**0.7758***



Modeling human observation & policy

- What if we want to predict the human's model state based on the human's observation dynamics?
- **Key insight**: learn differences from our *own* observation model (social projection)
- Apply relevant hyperparameters trained via RL

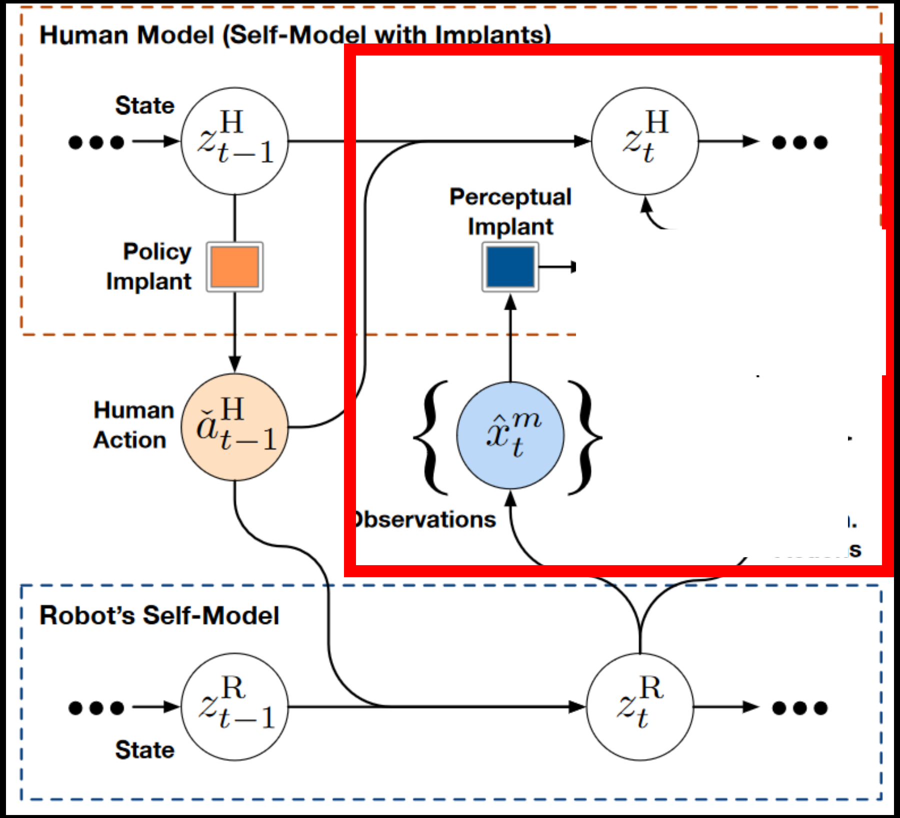




Chen, K., Fong, J., & Soh, H. (2022, June 27). MIRROR: Differentiable Deep Social Projection for Assistive Human-Robot Communication. Robotics: Science and Systems XVIII. Robotics: Science and Systems 2022. https://doi.org/10.15607/RSS.2022.XVIII.020

Social projection for human modeling (& model updating!)

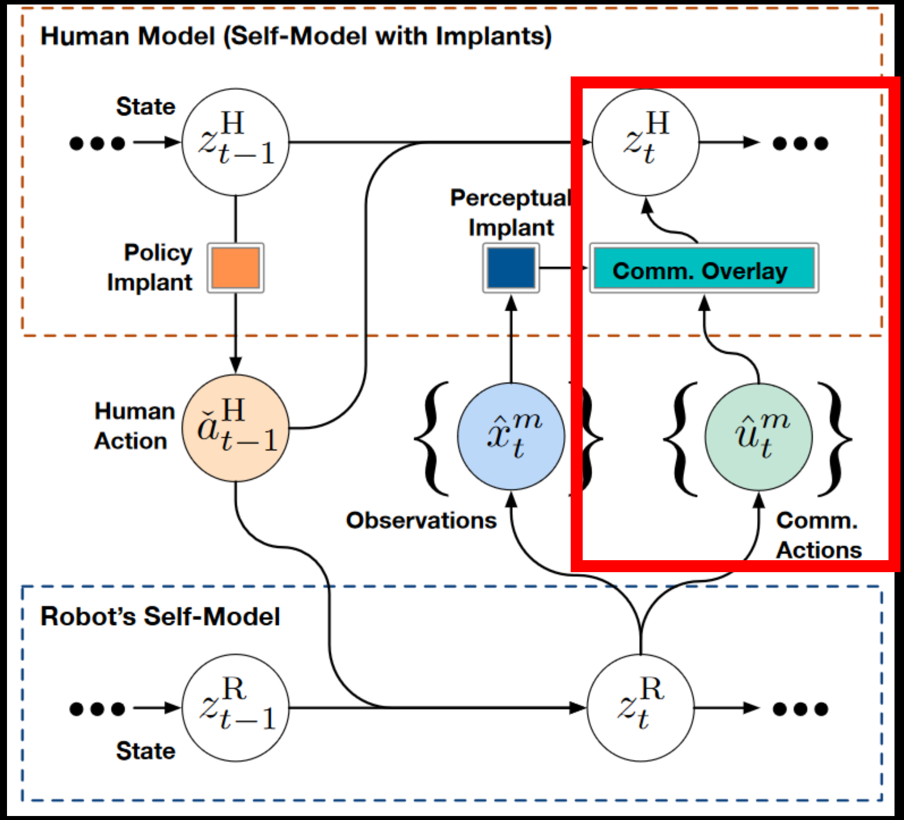
- With "perceptual implant," predict human's belief over state
- Use policy rollout to determine what model update will result in best reward, and by how much
- Deliver model update when reward outweighs communication cost



Chen, Fong, & Soh, 2023

Social projection for human modeling (& model updating!)

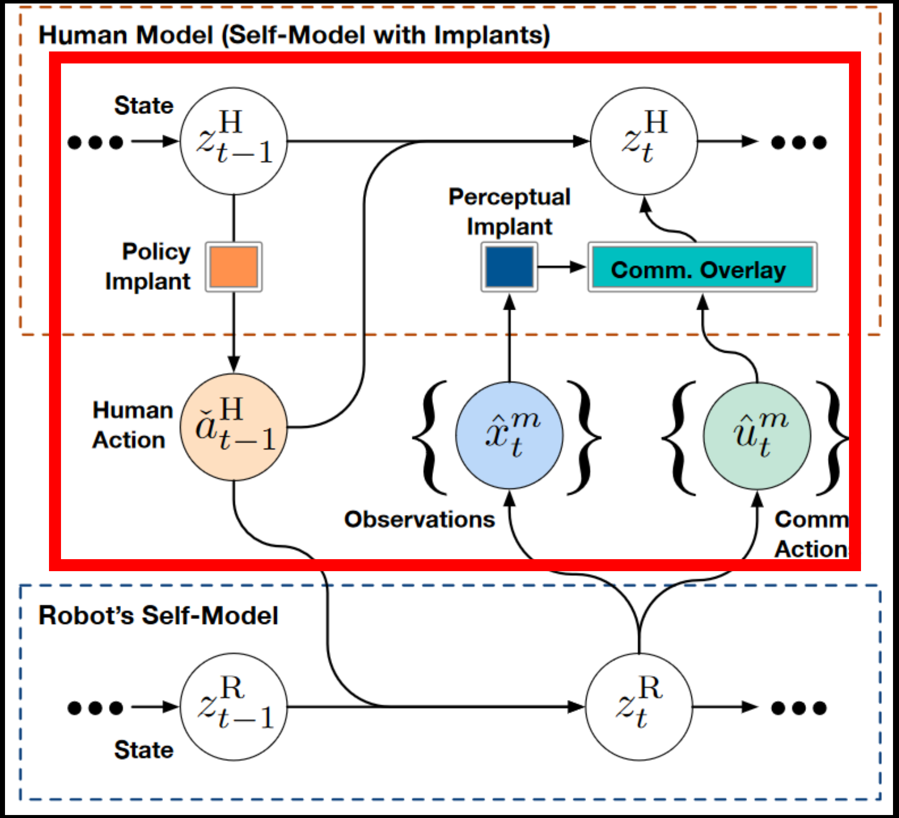
- With "perceptual implant," predict human's belief over state
- Use policy rollout to determine what model update will result in best reward, and by how much
- Deliver model update when reward outweighs communication cost



Chen, Fong, & Soh, 2023

Social projection for human modeling (& model updating!)

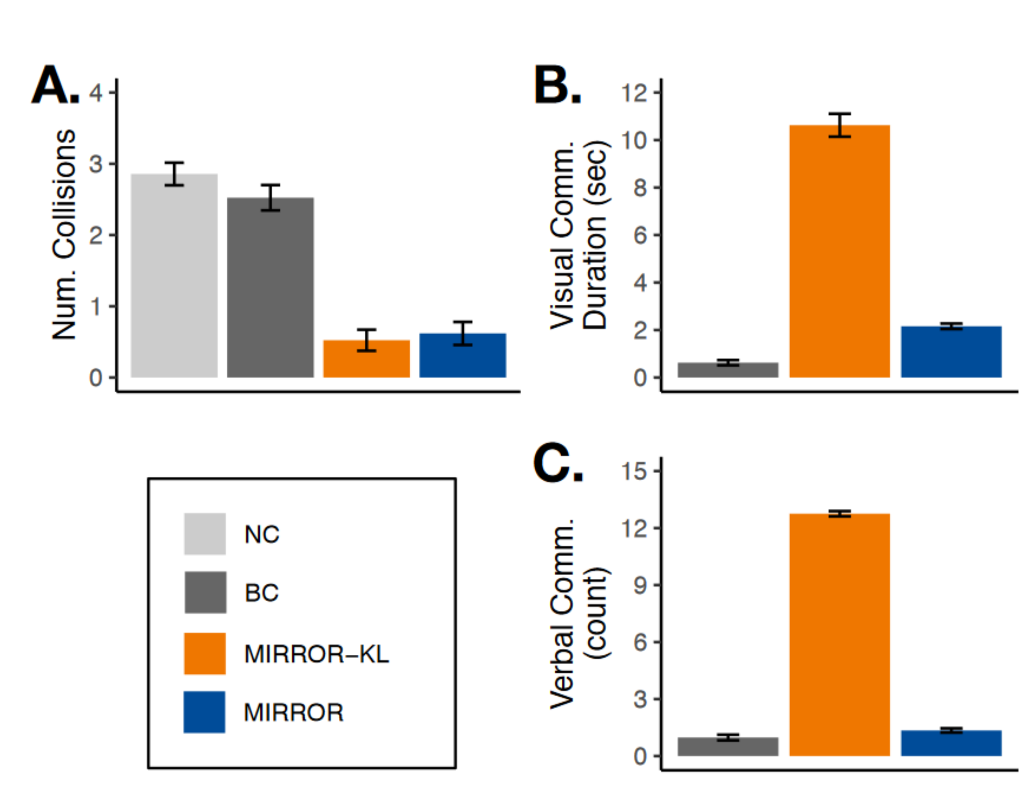
- With "perceptual implant," predict human's belief over state
- Use policy rollout to determine what model update will result in best reward, and by how much
- Deliver model update when reward outweighs communication cost



Chen, Fong, & Soh, 2023

Social projection for human modeling (& model updating!)

- Incorporating this human model into policy rollout results in fewer collisions with minimal communication

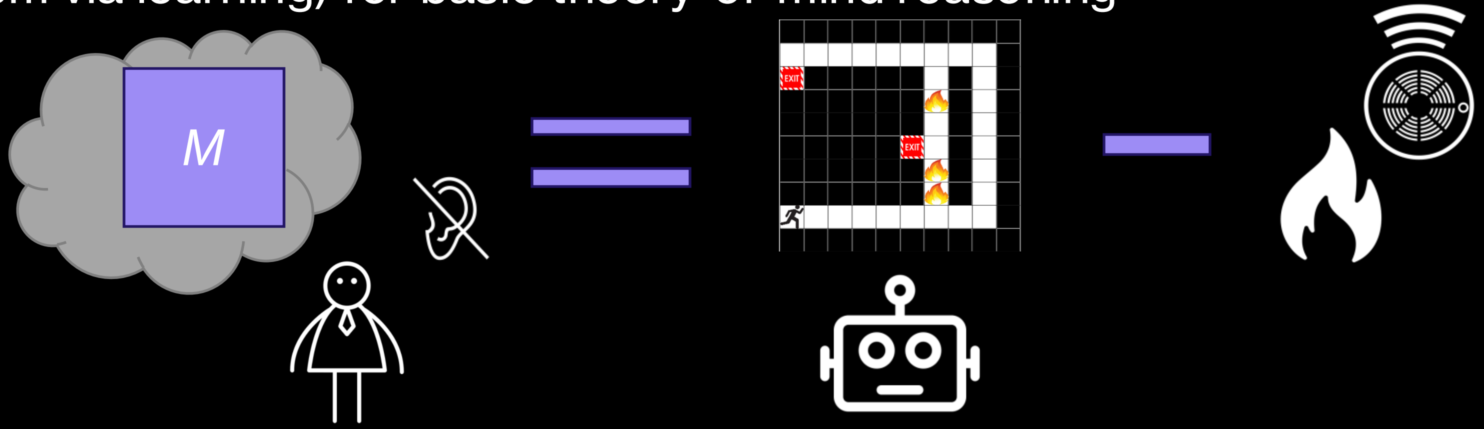


Chen, Fong, & Soh, 2023

Social projection for human modeling (& model updating!)

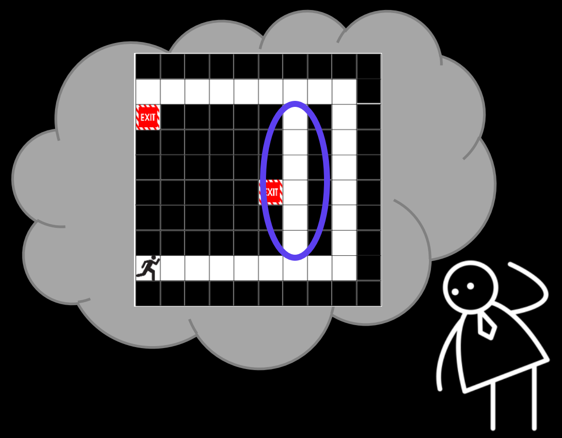
- **Key point(s):** By leveraging social projection, we can make reasonable estimations about human observations from our own, and use those estimates to predict the human model state (and update it appropriately)
- Incorporating how a model update will benefit the receiver is key to minimizing communication cost

- In non-driving domains, LLM interaction could help us augment our human models (rather than acquiring them via learning) for basic theory-of-mind reasoning



Recap: estimating receiver model state

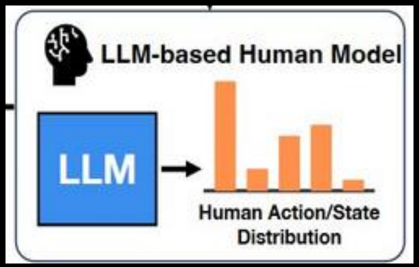
2. Estimate receiver model state (and model difference)

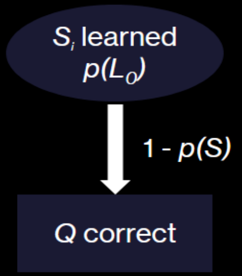


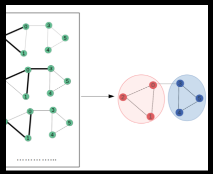
A. LLMs can act as human models for some applications

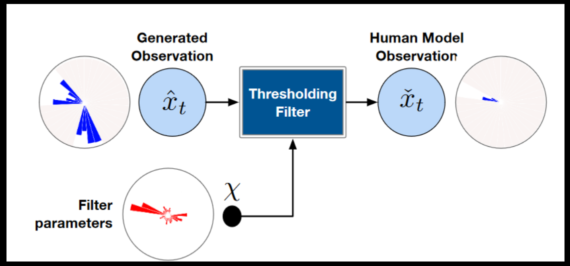
B. Knowledge tracing techniques can model human state from observed behavior
   - Can interact with user to improve estimate
   - Can learn from data to generalize and predict

C. Modeling human observation dynamics as modifications of the agent's can help predict model differences





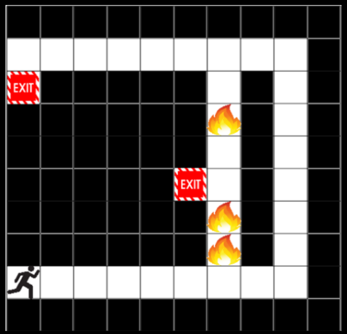




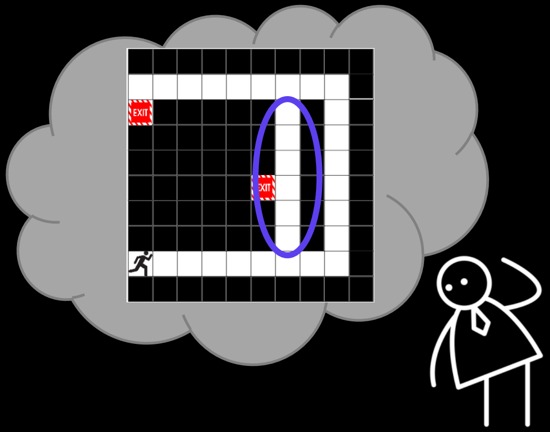
Tackling the component parts of handover

Current work & future directions

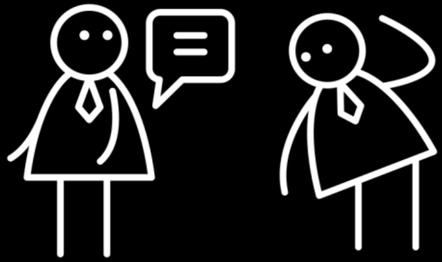
1. Form model state representation from input data



2. Estimate receiver model state (and model difference)



3. Generate handover communication for model update



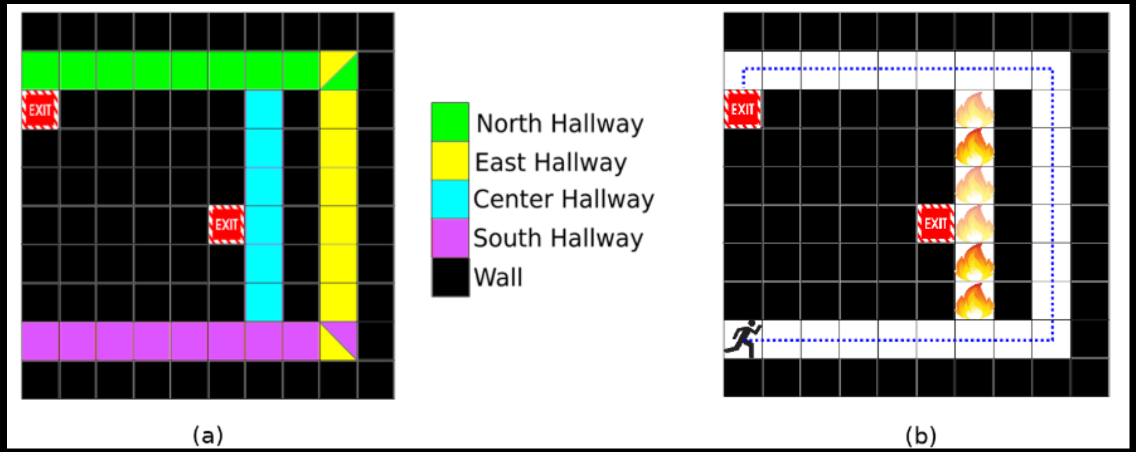
Model reconciliation via explanation

- Chakraboti et al. characterized some key explanation types for updating models, such as:
  - Plan Patch Explanation: generate and communicate an entire plan to the listener
  - Model Patch Explanation: communicate every model difference
  - Minimally Complete Explanation: shortest explanation that leads receiver to optimal plan.*
- Found in empirical studies that participants generally don't care about conciseness when receiving or generating updates -- but will if their bandwidth is explicitly limited
- This highlights the balance between computational challenges and HRI challenges

Chakraborti, T., Sreedharan, S., Zhang, Y., & Kambhampati, S. (2017). Plan Explanations as Model Reconciliation: Moving Beyond Explanation as Soliloquy (arXiv:1701.08317). ArXiv.
Chakraborti, T., Sreedharan, S., Grover, S., & Kambhampati, S. (2019). Plan Explanations as Model Reconciliation – An Empirical Study. 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI), 258–266.

Generating explanations for policy elicitation

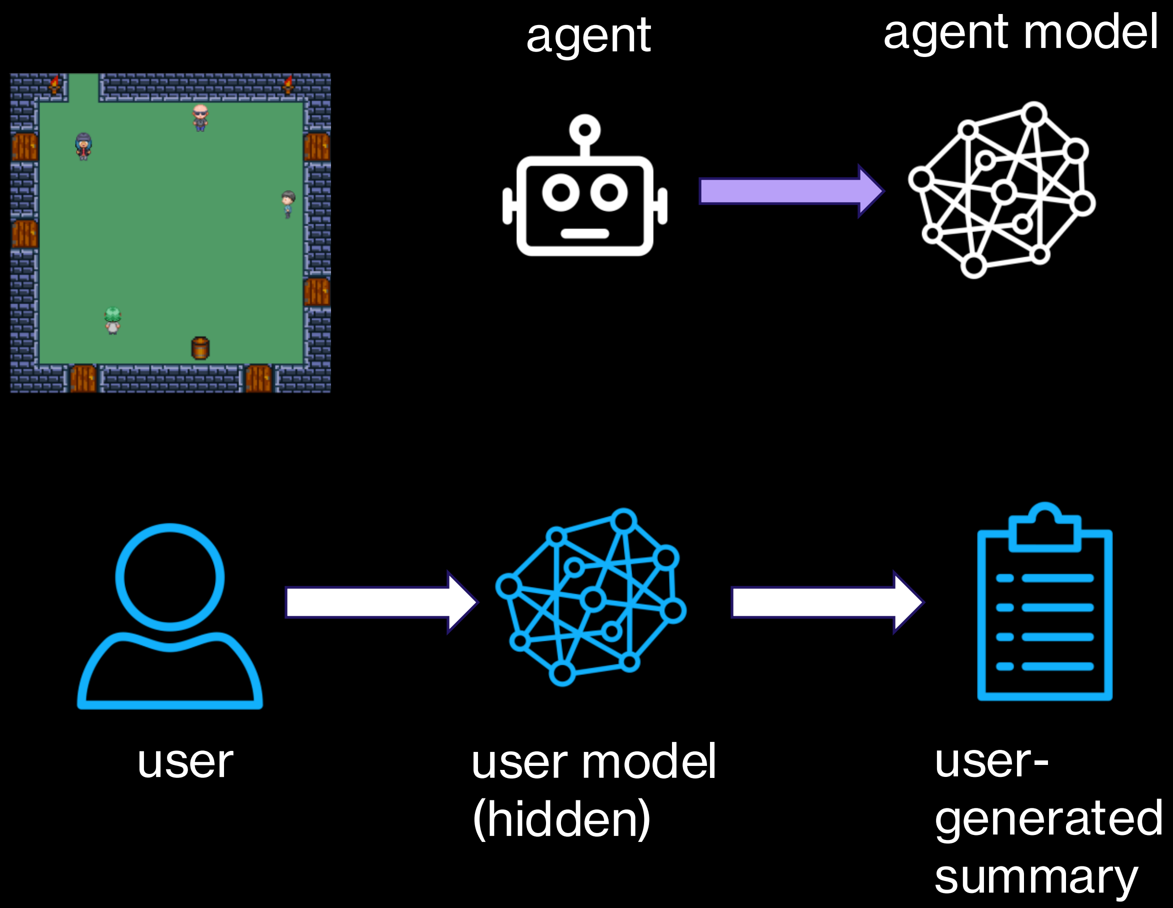
- Generate explanations to update user's reward function, with goal of eliciting a specific (optimal) policy
- Select states to update such that more optimal policy is achieved
- Treat explanation generation as a set-coverage problem using Boolean algebra, and optimize for conciseness



Tabrez, A., Leonard, R., & Hayes, B. (2021). One-shot Policy Elicitation via Semantic Reward Manipulation (arXiv:2101.01860). arXiv.

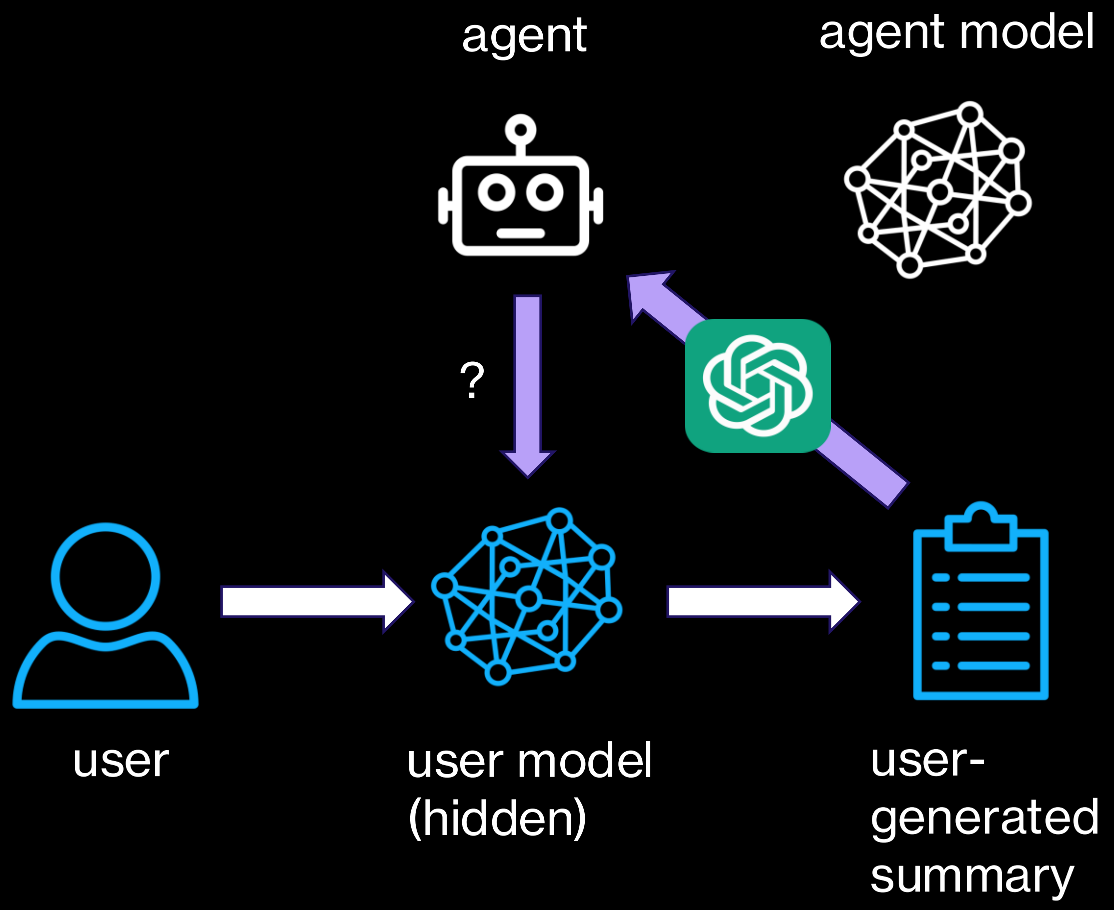
My current work: chatbot support of handover in multi-objective search

- Estimate world state from (incomplete) game telemetry
- Estimate human model state from their handover report via LLM
- Resolve uncertainty over both model state and relevance criteria via LLM-powered dialogue



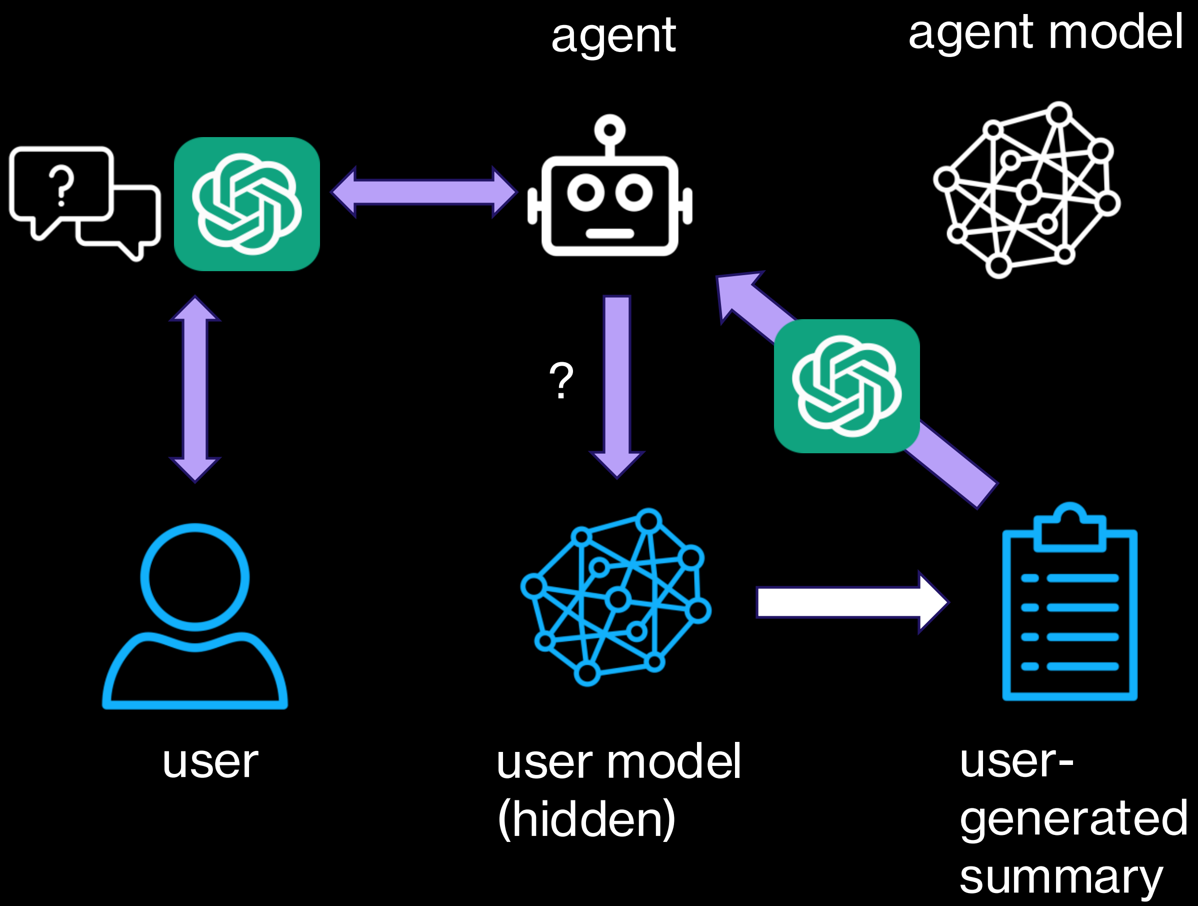
My current work: chatbot support of handover in multi-objective search

- Estimate world state from (incomplete) game telemetry
- Estimate human model state from their handover report via LLM
- Resolve uncertainty over both model state and relevance criteria via LLM-powered dialogue



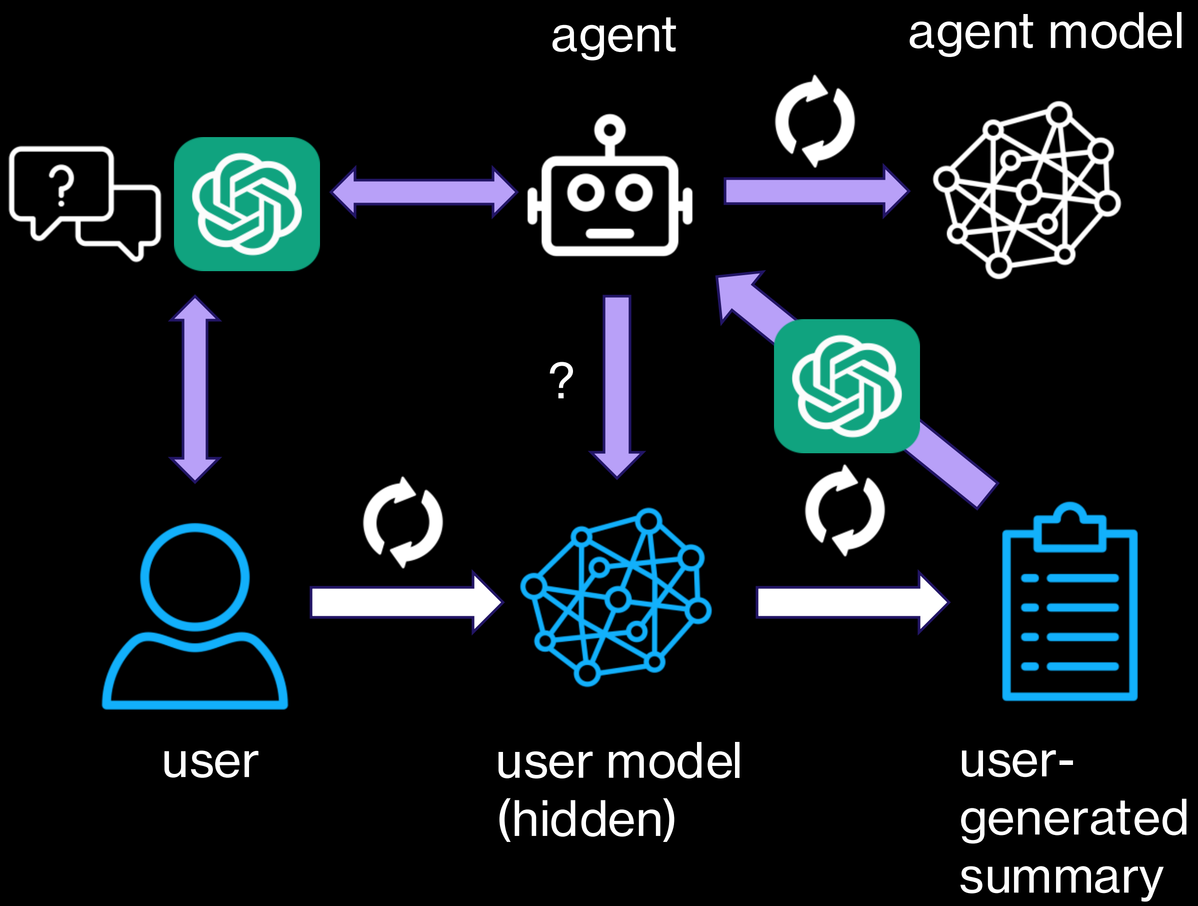
My current work: chatbot support of handover in multi-objective search

- Estimate world state from (incomplete) game telemetry
- Estimate human model state from their handover report via LLM
- Resolve uncertainty over both model state and relevance criteria via LLM-powered dialogue



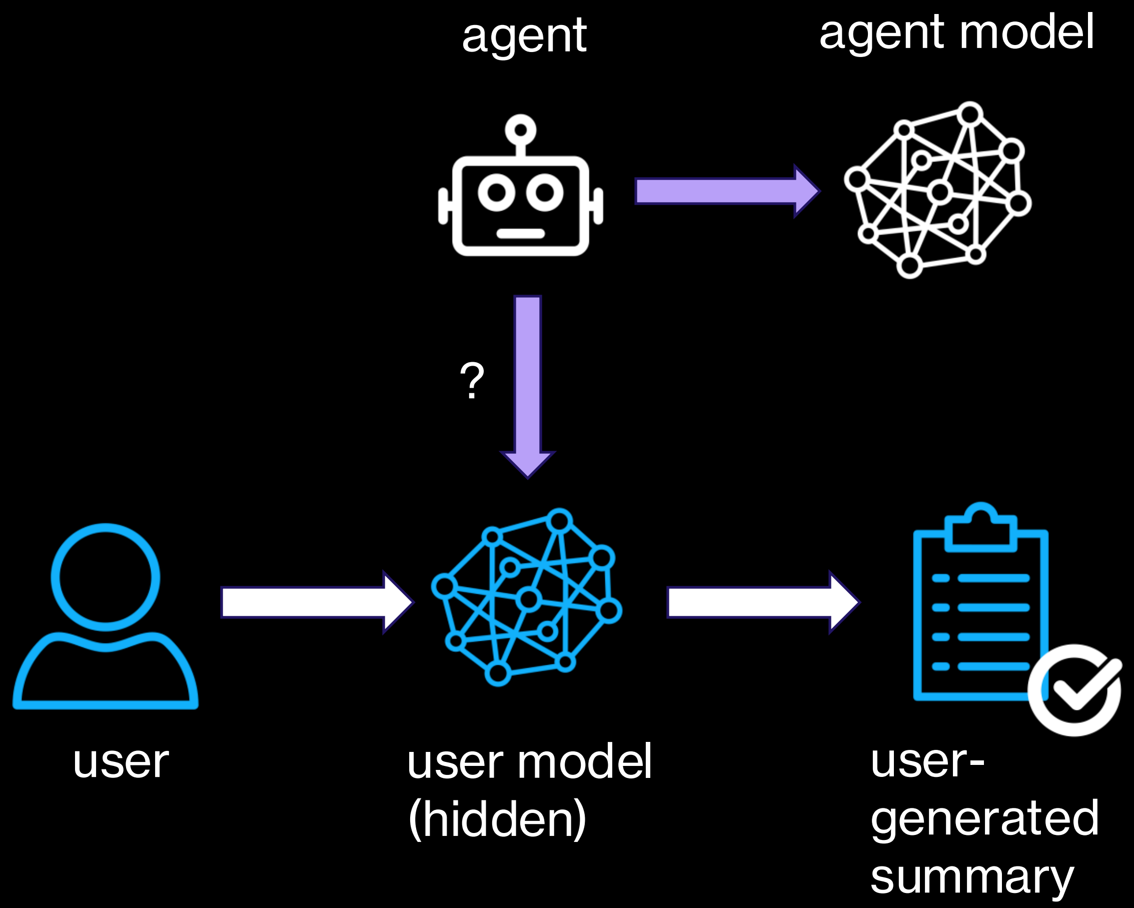
My current work: chatbot support of handover in multi-objective search

- Estimate world state from (incomplete) game telemetry
- Estimate human model state from their handover report via LLM
- Resolve uncertainty over both model state and relevance criteria via LLM-powered dialogue



My current work: chatbot support of handover in multi-objective search

- Converge to final summary
- Assess summary quality by utility / resulting task performance

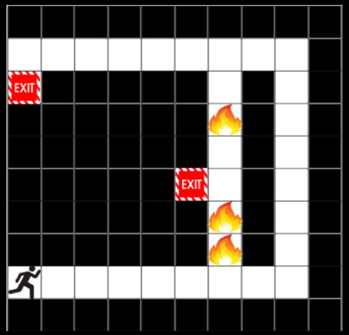


Why should AI participate in task handover?

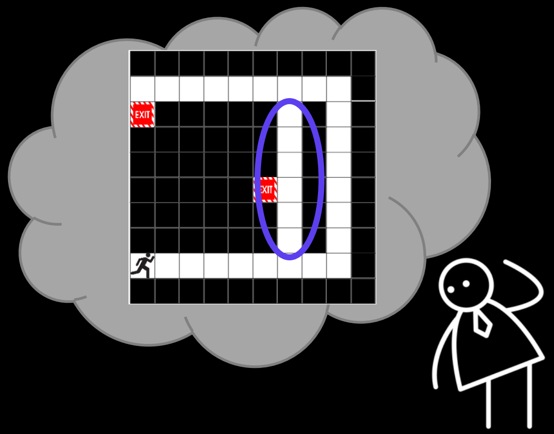
1. Handover is an important, yet underserved problem space
2. Recent advances in related areas of AI can be applied to task handover
3. LLMs have made these advances more applicable to real-world communication scenarios

Thanks for listening! Questions?

1. Form model state representation from input data



2. Estimate receiver model state (and model difference)



3. Generate handover communication for model update