

Characters

Lara J. Martin (she/they)

<https://laramartin.net/interactive-fiction-class>

Learning Objectives

Determine how task-oriented systems can be made with neural/ML-based methods

Tie task-oriented dialog back into IF/storytelling

Consider how modeling various aspects of characters affects a story

Speculate on how a system can be created to make rich characters

Review: Two Classes of Dialog Systems

1. Chatbots

- Systems designed for extended conversations
- Chatting for fun and entertainment

2. Task-Oriented Dialogue Agents

- Goal-Based Agents
- Siri/Alexa, interface with robots, booking flights or hotels

Review: Chatbots

Systems designed for extended conversations. Chatbots mimic unstructured conversations or 'chats' that are characteristic of informal human-human interaction

Architectures include:

Review: Frame-based Dialog Systems

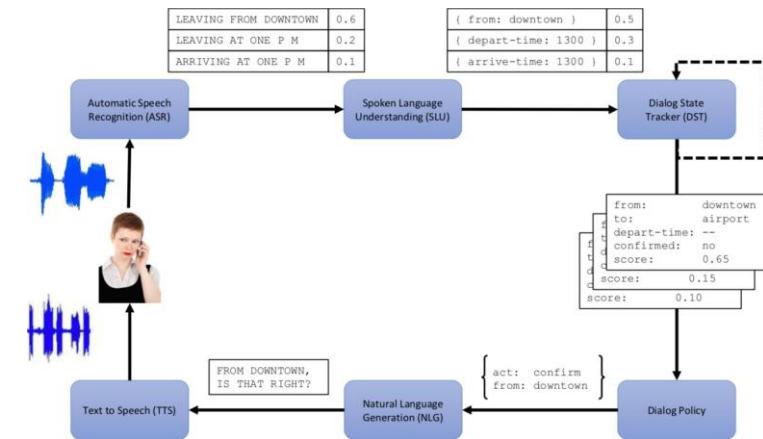
- Task-based Dialog Agents
- Based on “Domain Ontology”
 - A set of “Frames”
- Frame:
 - A knowledge structure representing user intentions
 - A collection of “slots”
 - Each “slot” having a set of “values”

■ Natural Language Understanding Component:

■ Dialogue State Tracker:

■ Dialogue policy:

■ Natural Language Generation Component:



Review: State Tracking Evaluation

Review: Dialog Policy Evaluation

End-to-end evaluation (Task Success)

Other potential metrics:

- Customer satisfaction (Survey questions)
- Length of conversation (Number of turns) – collecting all the information you need but not taking forever
- Relevance of response
- Accuracy of choosing the response similar to test data

Machine Learning for Slot Filling

- Supervised semantic parsing
- Model to map from input words to slot fillers, domain and intent
- Given a set of labeled sentences

“I want to fly to San Francisco on Tuesday”

Destination: SF Depart-date: Tuesday
- Requirements: Lots of labeled data (or perhaps an LLM?)

Slot Filling

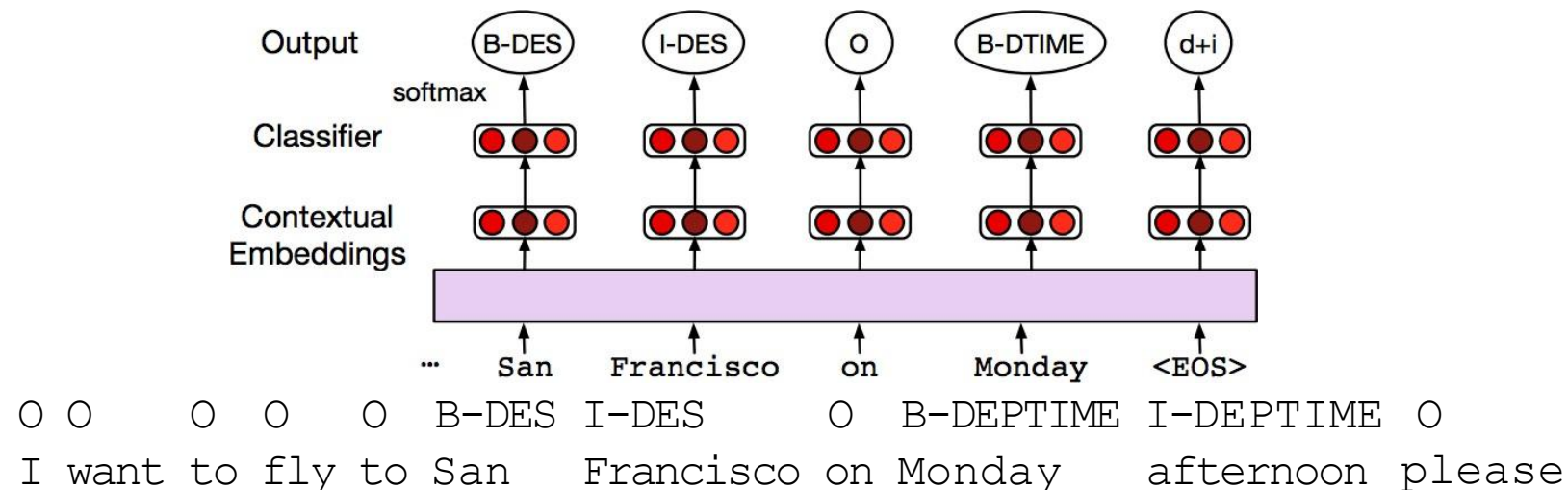
“I want to fly to San Francisco on Monday afternoon please”

Use 1-of-N classifier (Naive Bayes, Logistic Regression, Neural Network, etc.)

- Input:
features like word N-grams
- Output:
Domain: AIRLINE Intent: SHOWFLIGHT

More sophisticated algorithm for Slot Filling: IOB Tagging

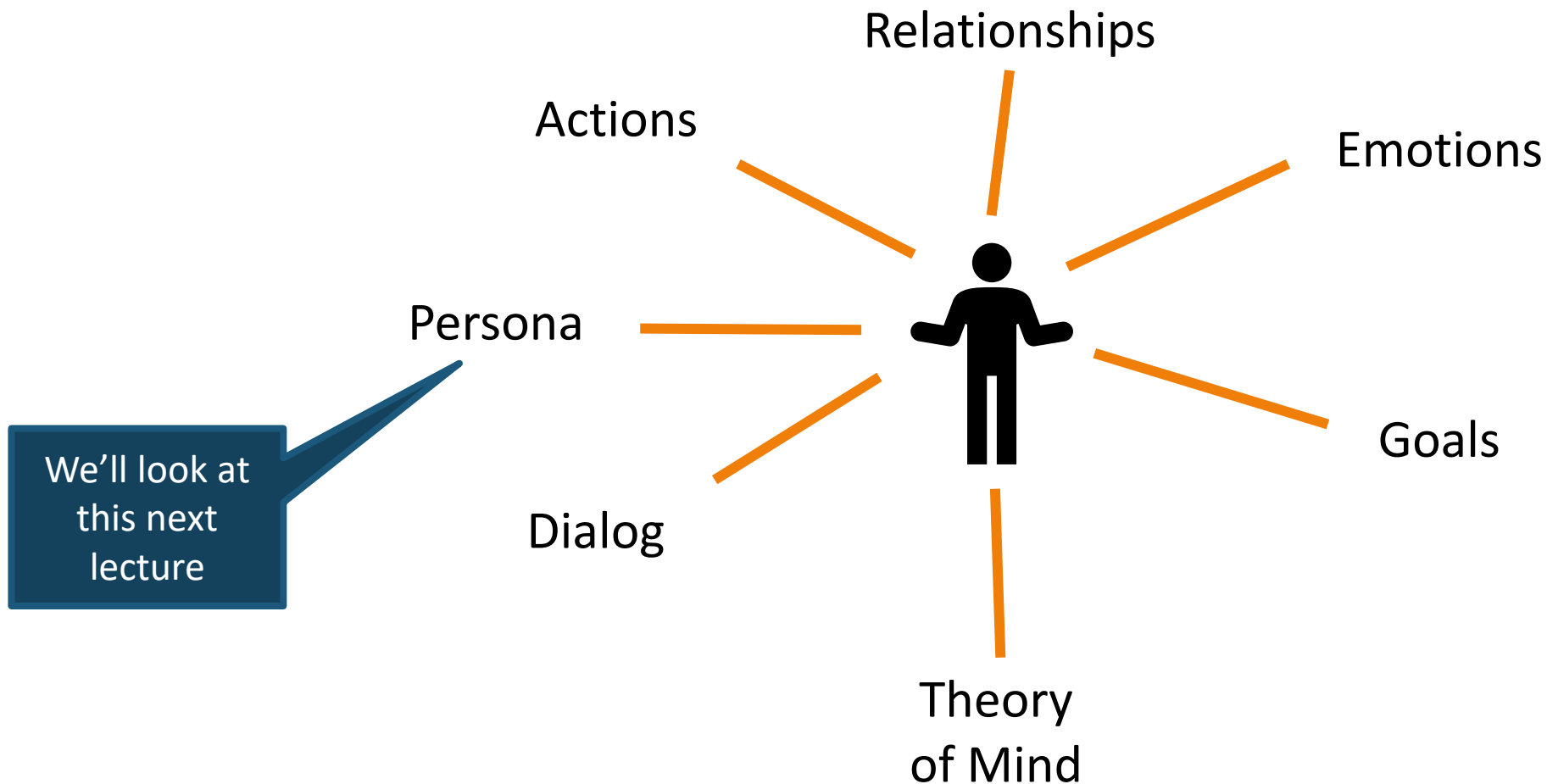
- IOB Tagging
 - Tag for the beginning (B) and inside (I) of each slot label,
 - plus one for tokens outside (O) any slot label
 - $2n + 1$ tags, where n is the number of slots
- Training Data: Sentences paired with sequences of IOB labels



Think-Pair-Share

How might **task-oriented** dialog be used in interactive fiction or storytelling?

What makes up a character?



What else might you want to model about a character?

Actions

Category:	Graveyard
Description:	Two-and-a-half walls of the finest, whitest stone stand here, weathered by the passing of countless seasons. There is no roof, nor sign that there ever was one. All indications are that the work was abruptly abandoned. There is no door, nor markings on the walls. Nor is there any indication that any coffin has lain here... yet.
Backstory:	Bright white stone was all the fad for funerary architecture, once upon a time. It's difficult to understand why someone would abandon such a large and expensive undertaking. If they didn't have the money to finish it, they could have sold the stone, surely - or the mausoleum itself. Maybe they just haven't needed it yet? A bit odd, though, given how old it is. Maybe the gravedigger remembers... if he's sober.
Neighbors:	Dead Tree, south, following a dirt trail behind the mausoleum Fresh Grave, west, walking carefully between fallen headstones
Characters:	gravedigger, <i>thief</i> , <i>peasant</i> , <i>mouse</i> , <i>bat</i>
Objects:	wall, <i>carving</i> , <i>leaf</i> , <i>dirt</i>

(a) Example room created from the room collection and labelling tasks.

Actions

Query:	chicken	pirate	coffin	rake	tavern	meadow
objects	chicken coop eggs a pen for the chickens chimney corn	Pirate swords dock cargo ship seagulls on the dock	the remains remains bones bones of the innocent adventurer's remains	shovel garden a garden Hand carved stone garden bench	Ale bottles beer mug of mead a large ornate table beer keg	flower pot fruit An enchanted amulet. citrus fruit fruit trees
characters	chickens fox trying to steal chickens farmers The farmers farmer	boat captain captain merchant boat workers workers	spirits of our ancestors mourner zombies families bandit	gardener stable hand Garden dog stable boy A stable boy	tavern owner bartender Goblin King's bartender A serving wench Serving wench	a deer a songbird fruit bats parent butterfly
locations	Chicken Pen Corn field Farmer's house Large Farm Pig Pen	Pirate Ship Dock at the Port Loading Dock Fishing Dock crew berthing	Old Crypt sacristy Disposal area inside temple crypt Sacrifice Chamber	Across the King's Garden Hidden garden The garden courtyard Church garden Tool Shed	The werewolves tavern Tavern of Browntavia Port Tavern The bar bazaar outside the royal city	Lush meadow Flower Field flower garden Mushroom Hut Archery zone
actions	get chicken hug chicken hit chicken give cowbell to chicken steal sword from chicken	hug pirate hit pirate steal sword from pirate steal cargo from pirate give cargo to pirate	put torch in coffin get torch from coffin put bone in coffin get bone from coffin hit archaeologist	get rake drop Rake steal Rake from gardener give Rake to thing give Rake to person	hug tavern owner give food item to tavern owner give telescope to tavern owner drink drink drop drink	get flower from meadow put flower in Meadow give Flower to a deer give Flower to deer steal Flower from a deer
vocabulary	bock tasty bawk moo egg	crew ye port sea seas	archaeologist robber crypt loss adventures	vegetable carved alice hook exorcisms	drink drinks regular item tip	flower amulet songbird wasp an

Table 3: Neighboring Starspace phrase embeddings (no pretraining from other data) for different types of entities and actions. The first row are arbitrarily chosen queries (chicken, pirate, coffin, rake, tavern, meadow), and the subsequent rows are their nearest objects, agents, locations, actions and vocabulary in embedding space.

Actions

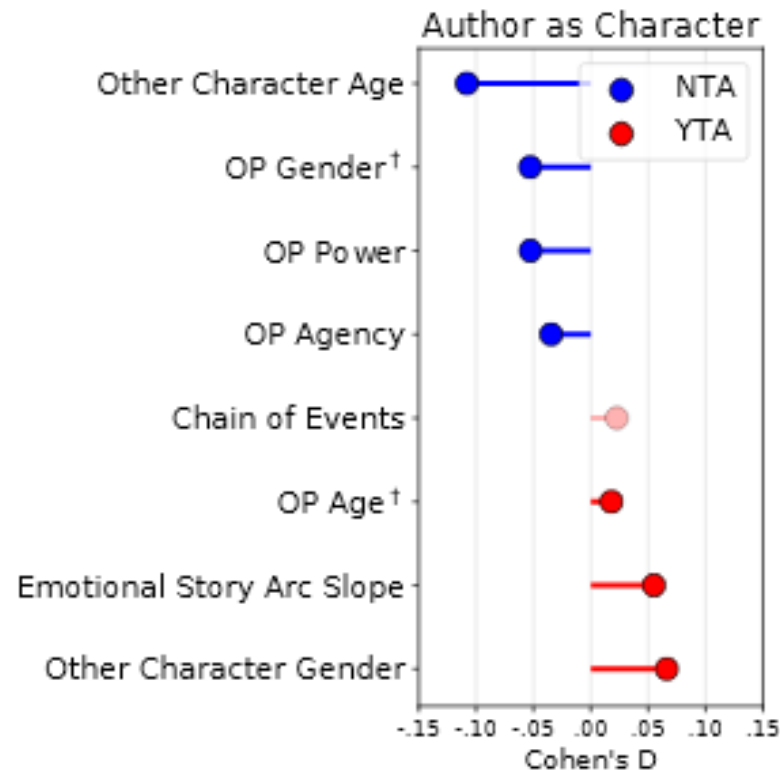


“The currently played role fills in once the player initiates conversation. It has information regarding the character’s personality, profession, age, gender, marital status, physical appearance, and their reason for being at the current location (work, errands, leisure, etc.)”



“When not updating the simulation, the wizard has time to explore the history of the town and the interweaving relationships of its denizens. When he unearths narratively interesting tidbits, he communicates them to the actor via a chat window. “

Actions



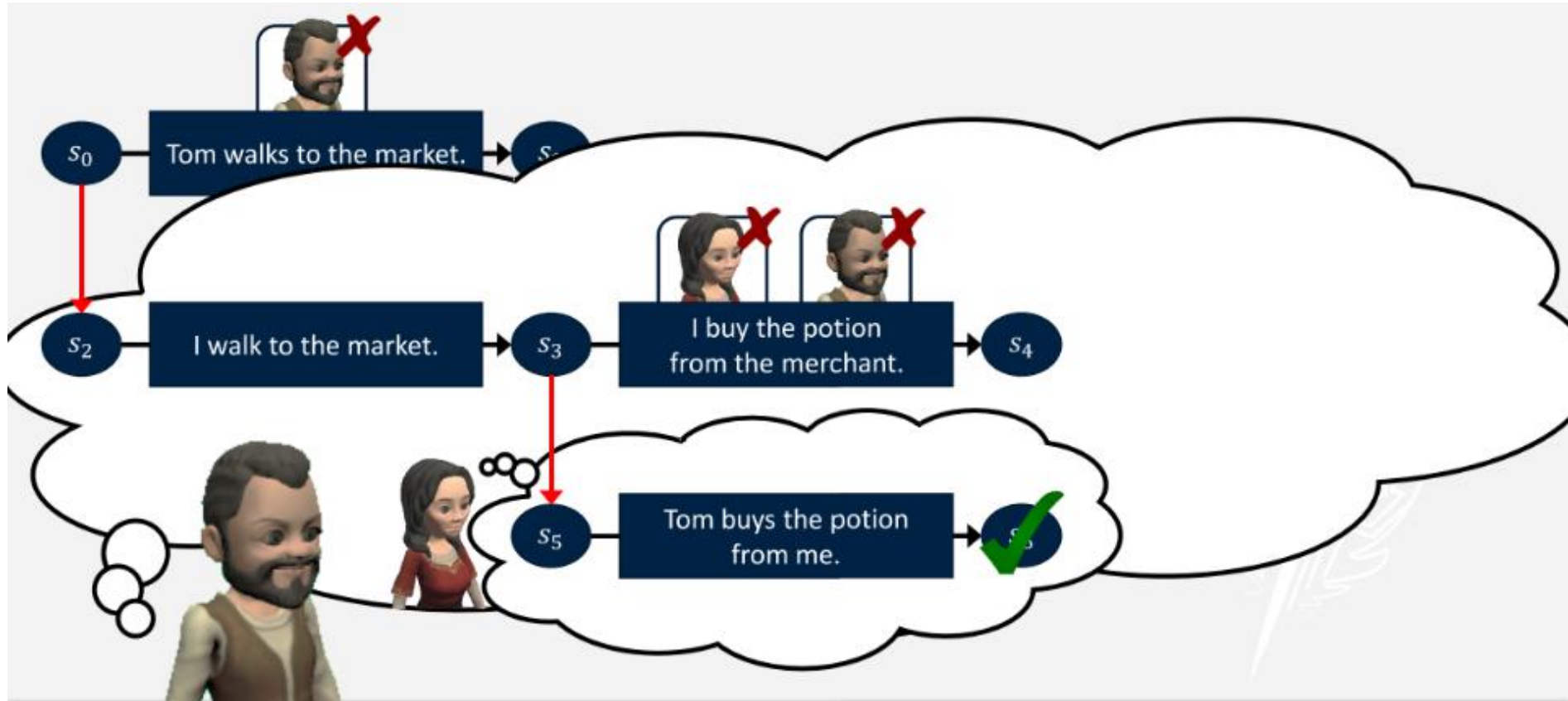
(a) Cohen's D values showing the correlation of features for YTA & NTA classes. Lighter shaded points are not significant at a with Benjamini-Hochberg corrected significance $\alpha < 0.05$). The higher the absolute effect size, the more that feature is associated with the YTA/NTA class. † includes a binary covariate equal to 1 for undisclosed age/gender when calculating significance via the logistic regression (Cohen's D is a bivariate measure and, thus, unable to account for this covariate).

Actions

Control Feature	Description	Expected Impact on Model's Output
Player ID	Player writing a given dialog turn	Connects the current turn to the player's previous turns, which is important in multi-party conversations.
IC versus OOC	Whether a player is in-character or out-of-character for a given dialog turn	Changes whether the generated text is more like descriptive text found in a novel, or more like a discussion of rules and strategies.
Character Name	Name of the character being played by the player of a given dialog turn	IC descriptions use the character's name.
Character Class	D&D classes	Character classes perform different actions (e.g. wizards cast spells, thieves pick locks)
Character Race	D&D fantasy races	Different physical characteristics (e.g. halflings are small, dragonborn have scales).
Character Pronouns	The character's pronouns	Uses the correct pronouns when describing the character.
Character Actions	List of actions taken by the character in the current turn	Allows a description to be generated for a given action. The action can be thought of as a goal for the description.
Combat	Whether the players are currently engaged in combat or not during a given dialog turn	Affects the likelihood of actions (e.g. attacks are more likely during combat and investigations checks are more likely outside of combat)

Table 2: Our LLMs are conditioned on a variety of control features that allow the models to better learn what kind of text to generate for the next utterance prediction task

Theory of Mind (ToM)



Theory of Mind (ToM)



Definition (Belief State). Given a world frame $W = \langle GL, C \rangle$, a belief state for some character $c \in C$ is a tuple $BS_c = \langle B_c^+, B_c^-, U_c \rangle$ such that B_c^+, B_c^- and U_c together form a partition of GL , where B_c^+ designates all the ground literals that c believes to be true, B_c^- includes all the ground literals that c believes to be false and U_c designates all the ground literals that c does not believe to be true and does not believe to be false.

Figure 1: A solution plan for the Drink Refill domain's planning problem. Green actions are successfully performed actions. Red actions are ones that are attempted but that fail because their material preconditions are not all met in the world state where they are attempted.

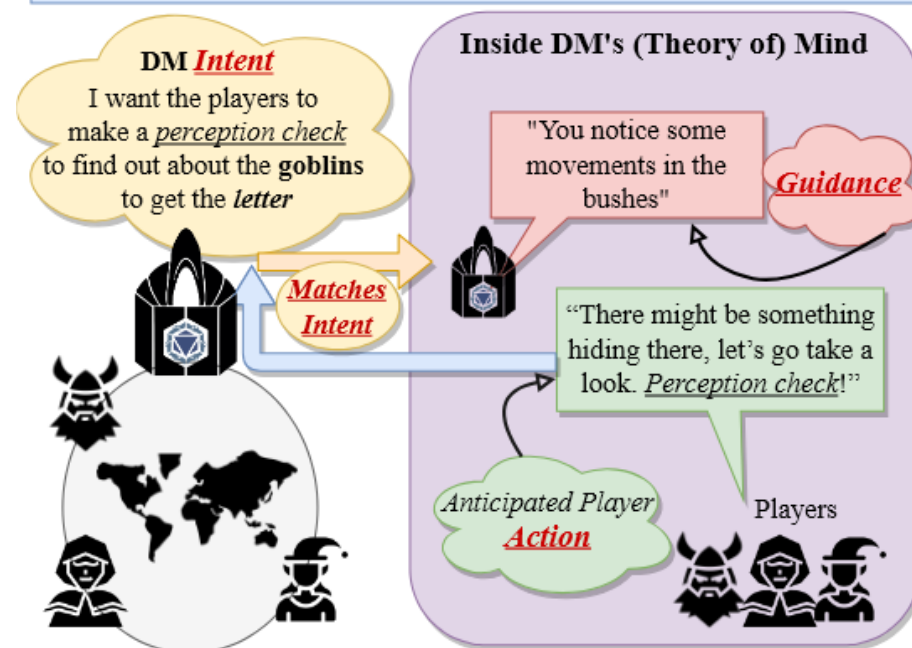
Theory of Mind (ToM)

Shared Common Ground between the DM and Players

Players were hired by a dwarf named Gundren Rockseeker to transport a wagonload of provisions to Phandalin. After a day and a half of travel, the players got onto a smaller trail not as well maintained...

Information Only Available to the DM

Five **goblins** hid in the bushes near the trail ready to attack the players. Upon defeating them, players can find a **letter** from one of the goblin's pockets showing that Gundren has gone missing...



Theory of Mind (ToM)

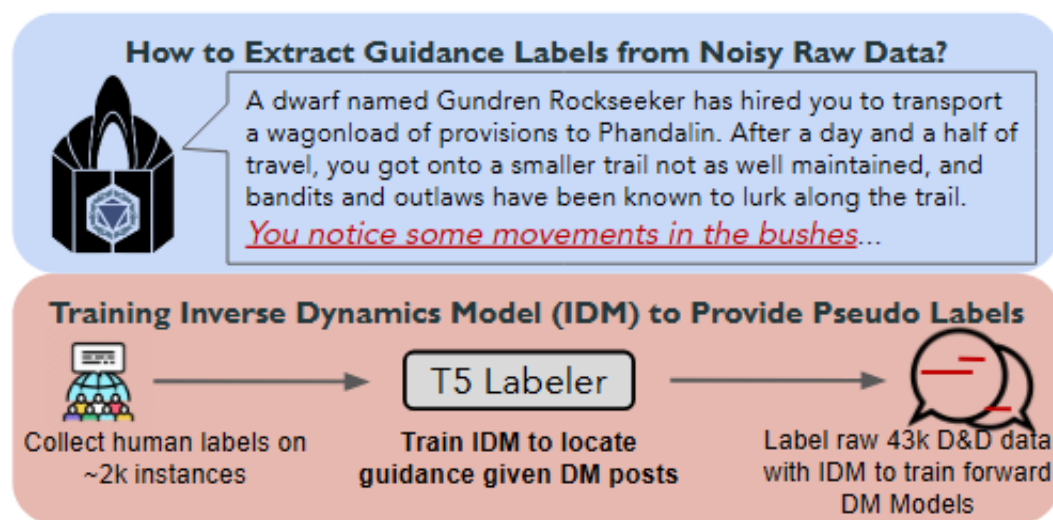


Figure 2: Illustration of IDM. We collect 2.5k human labels on guidance and train an IDM labeler to generate pseudo labels for unlabeled large corpus.

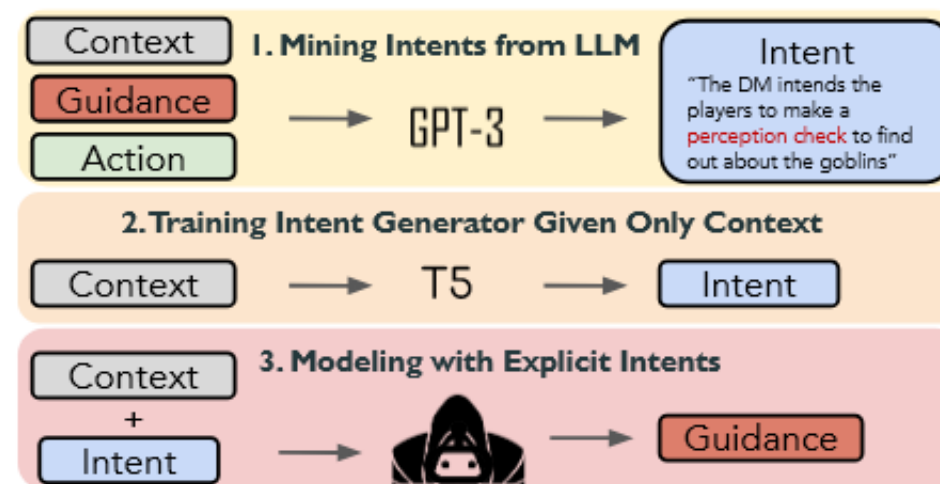
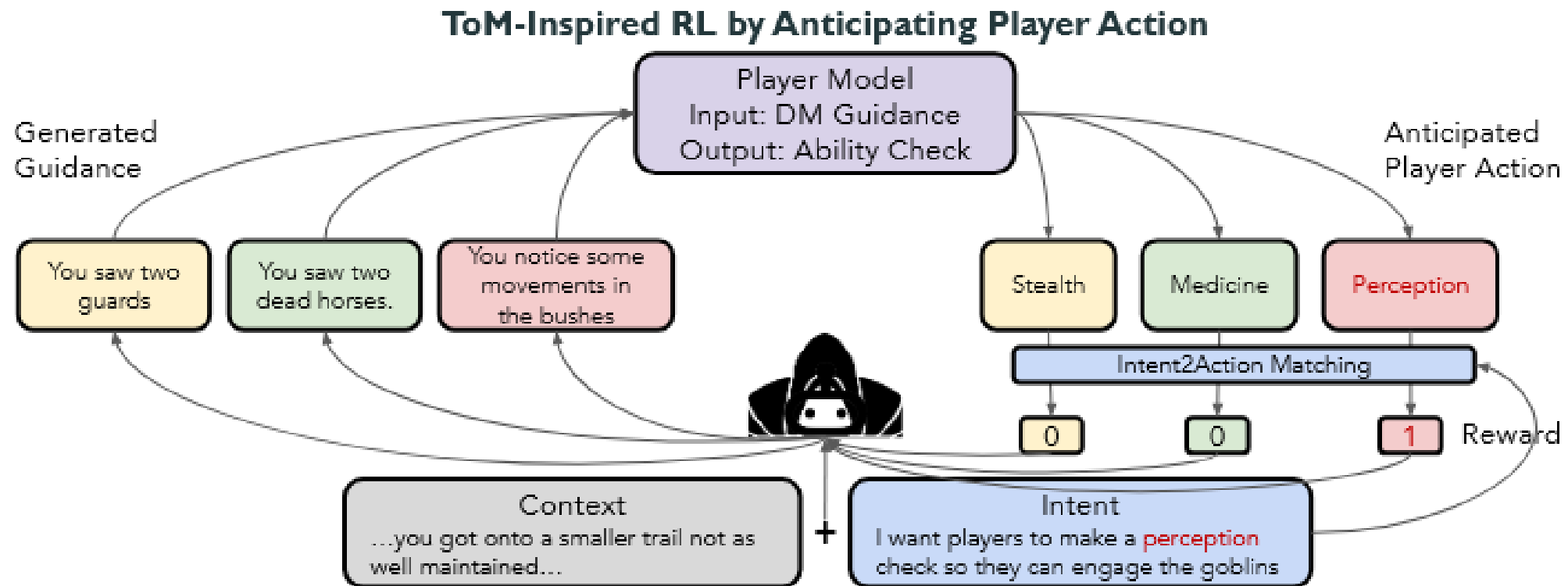


Figure 3: Illustration of intent modeling. We first mine intents from LLM and then train an intent generator to generate intent as additional context to train the DM model.

Theory of Mind (ToM)



Theory of Mind (ToM)

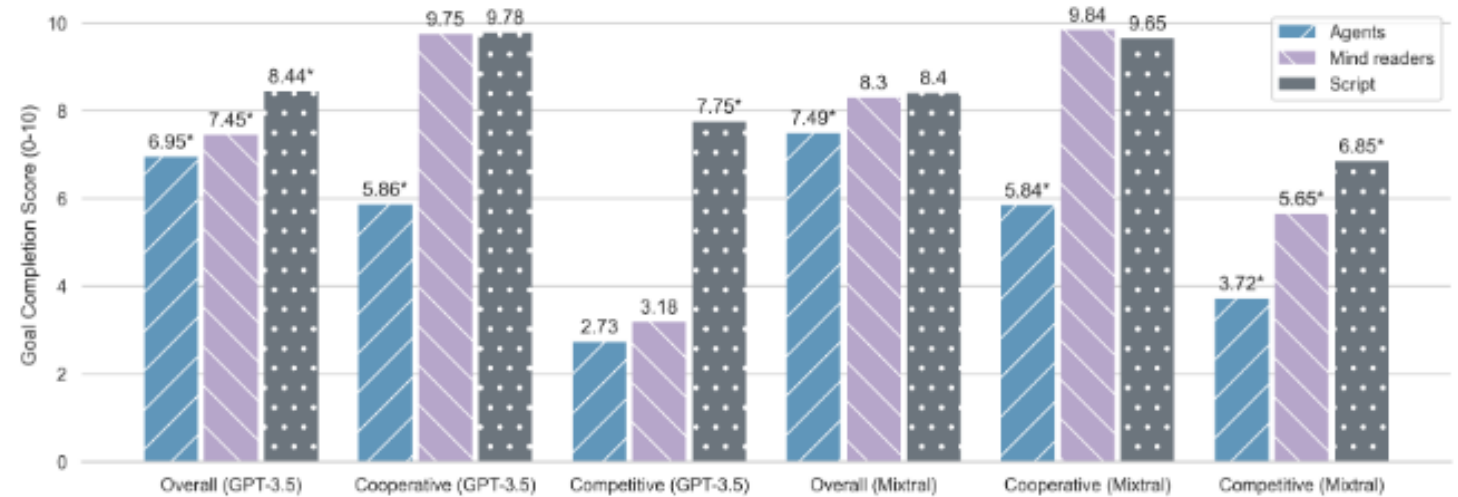
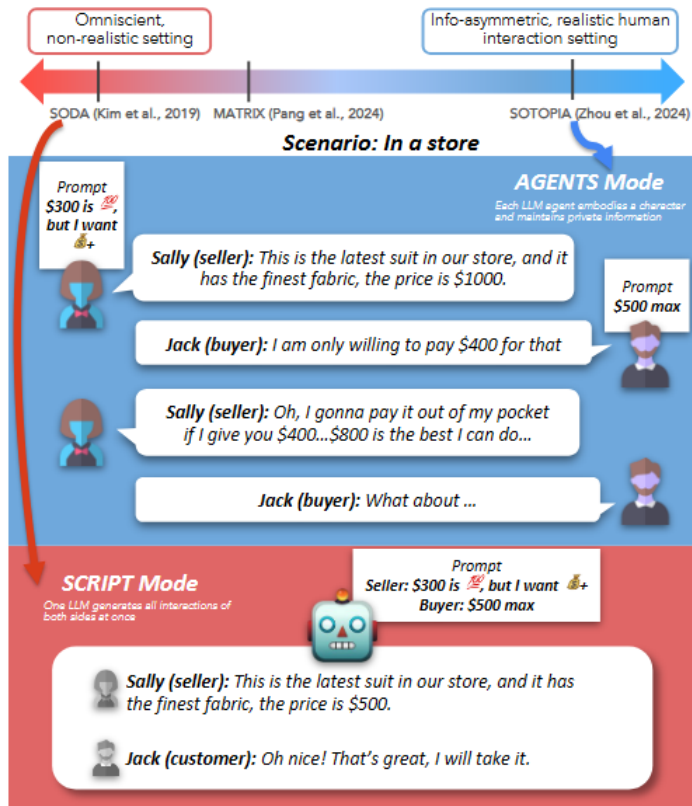


Figure 2: Average goal completion score of models across different modes in various settings. Overall contains all the scenarios, and the other two contains representative scenarios from the cooperative and competitive scenarios. We perform pairwise t-test, and * denotes the score is statistical significantly different from the other two modes in this setting ($p < 0.001$).

Goals

Self: guard Partner: archer	Self: swimmer Partner: turtles
Persona: I guard the castle. I guard the king. I would kill to protect the royal family	Persona: I am a huge fan of deep sea exploration, but I take any chance I can get to go for a swim...
Setting: The armory, Inside Tower. The near top of the tower 6 feet before the very top. Where the watchers keep their eye...	Setting: Bank, Swamp This is a grassy area that surrounds much of the swamp. It's a plain field with some trees nearby along...
U_0^{player} This is the armory! The king keeps the best weapons here. Take a look -	U_0^{player} Just keep taking good care of your beautiful little turtle family! Your species is quite unique and I love to see you about when I go for a swim.
U_0^{env} Hello, I need to get into the palace to see the king. I think he might like to see these weapons.	U_0^{env} Well, thank you for that. Do you happen to know where my other turtle friend is? You haven't captured any turtles have you?
A_0^{env} get weapon	A_0^{env} hug swimmer

Table 2: Example 1-step episodes where after the Topic RL agent's utterance U_0^{player} the environment agent's response action A_0^{env} was equal to the RL agent's goal g . Our RL agent both makes natural utterances given the situation, and that elicit the desired goal.

Goals

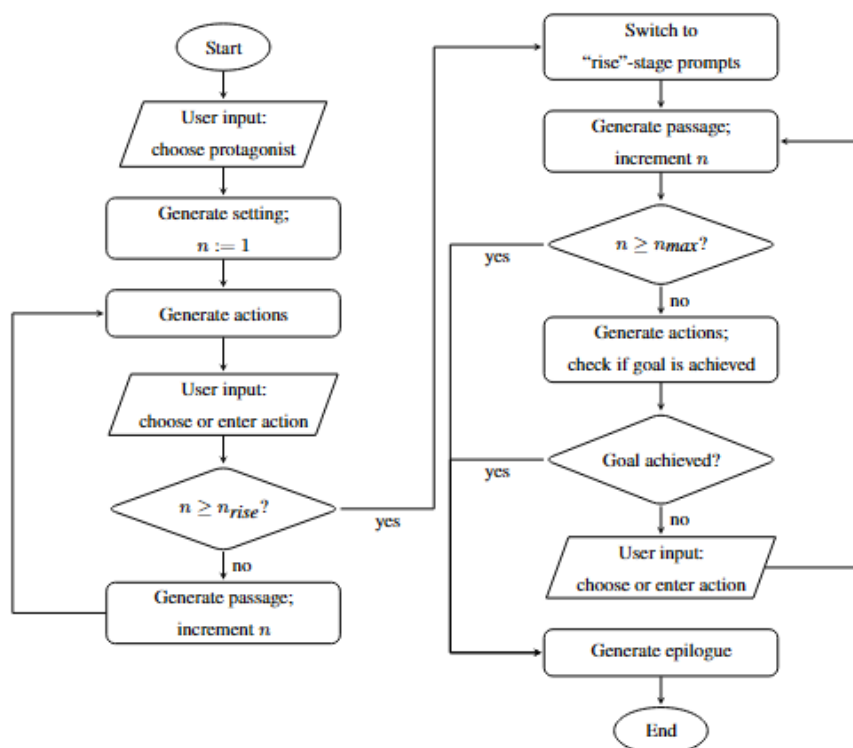


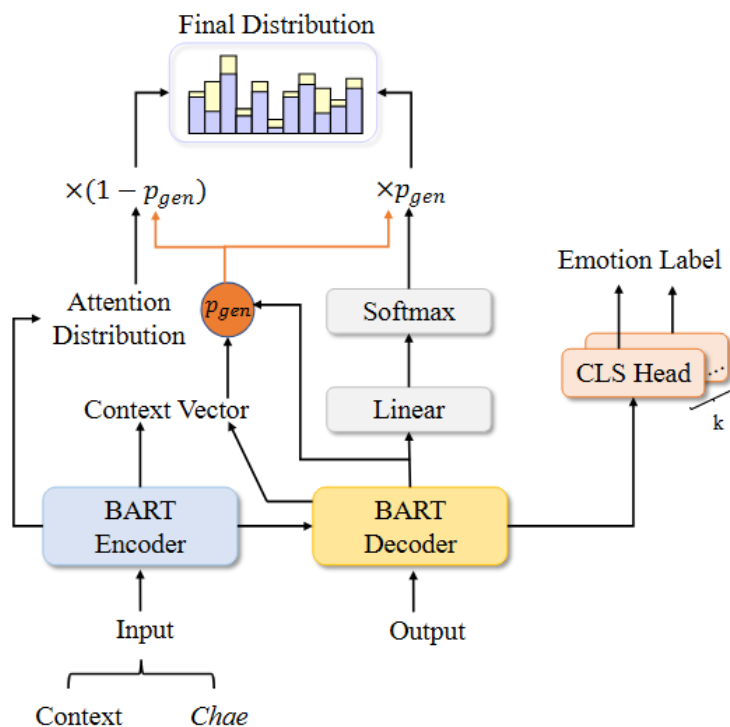
Figure 4: Story generation workflow. The left-hand side corresponds to the “low” stage of the story, the right-hand side to the “rise” stage.

You are a language model for writing WHOLESOME children’s fairy tales suitable for six-year-olds [...] The protagonist of the fairy tale is {name}. Their goal is to {goal}. The child will submit an action undertaken by the protagonist, and you will write the next plot point of the story [...] Your answers develop the plot and logically follow from the protagonist’s actions. However, the protagonist always faces challenges and NEVER reaches their goal [...]

You are a language model for writing WHOLESOME children’s fairy tales suitable for six-year-olds [...] The protagonist of the fairy tale is {name}. Their goal is to {goal}. The child will submit an action undertaken by the protagonist, and you will write the next plot point of the story. [...] Your answers develop the plot, logically follow from the protagonist’s action, and bring them closer to their goal [...]

Figure 2: System prompt templates for passages in the “low” (left) and “rise” (right) stages of the story. Placeholders for story-specific information are highlighted in red

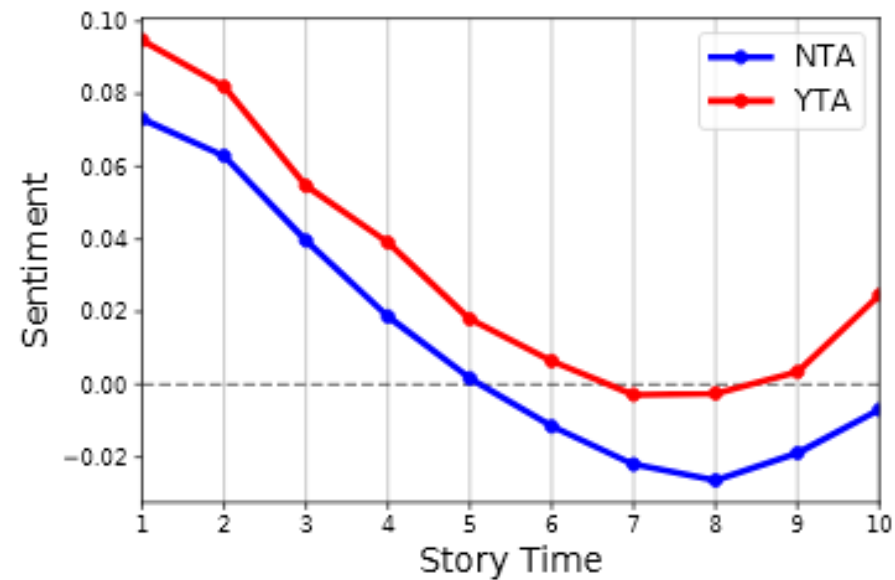
Emotions



Context	A polite thief was making robberies in the small town.
Chae1	$\langle SEP \rangle \langle soc \rangle$ People $\langle soa \rangle \langle no_action \rangle \langle soe \rangle$ fear $\langle SEP \rangle \langle soc \rangle$ Man $\langle soa \rangle$ to catch the thief $\langle soe \rangle$ anger
Result1	One day, a man walked up to him and asked him to stop .
Chae2	$\langle SEP \rangle \langle soc \rangle$ People $\langle soa \rangle \langle no_action \rangle \langle soe \rangle$ fear $\langle SEP \rangle \langle soc \rangle$ Man $\langle soa \rangle \langle no_action \rangle \langle soe \rangle$ joy
Result2	The man who was supposed to stop him was a nice man .
Chae3	$\langle SEP \rangle \langle soc \rangle$ People $\langle soa \rangle \langle no_action \rangle \langle soe \rangle$ fear $\langle SEP \rangle \langle soc \rangle$ Tom $\langle soa \rangle$ to catch the thief $\langle soe \rangle$ anger
Result3	Tom decided to investigate and caught the thief .
Chae4	$\langle SEP \rangle \langle soc \rangle$ People $\langle soa \rangle$ call the police $\langle soe \rangle$ fear $\langle SEP \rangle \langle soc \rangle$ Tom $\langle soa \rangle$ call the police $\langle soe \rangle$ anger
Result4	Tom called the police and they told him to call the police .

Table 6: Case study of controllability.

Emotions



(b) Emotional Story Arc. Average VADER sentiment across 10 equally-sized sentence-level chunks. Positive values are positive sentiment, negative values are negative sentiment.

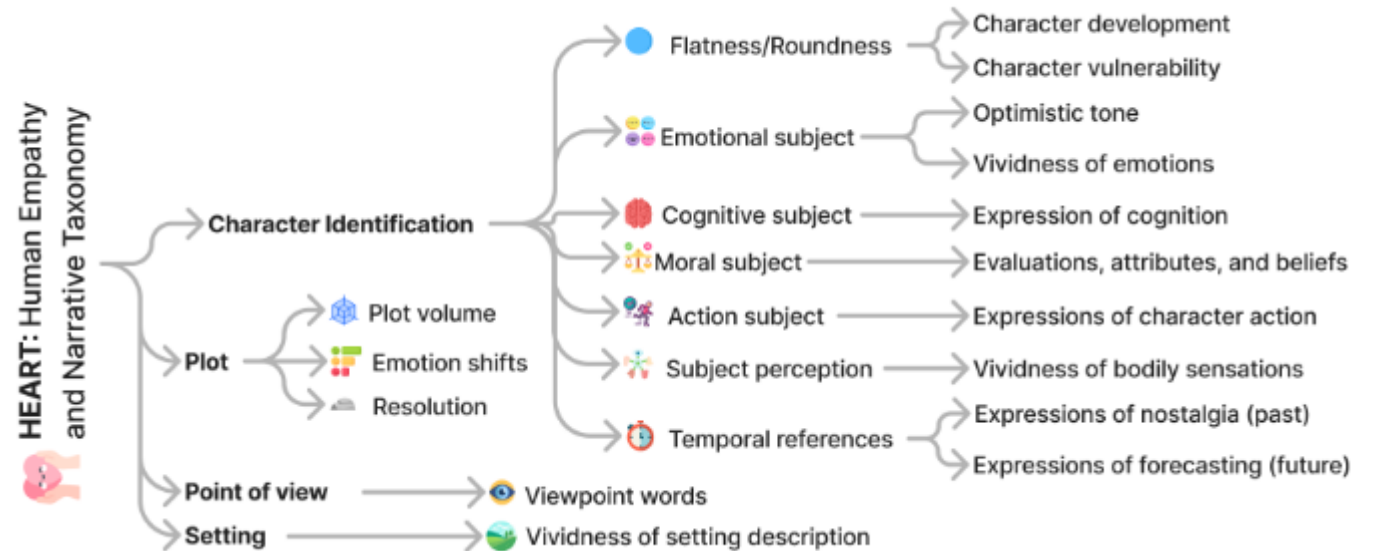
Emotions

Story: It was a long and difficult pregnancy. I felt like my insides were being ripped apart. But at 4:15 pm, I gave birth to a beautiful baby. I was totally exhausted, with cold tears streaming down my face. But looking into my baby's eyes, all the pain disappeared, and I just felt warmth in my heart.

Narrative Elements



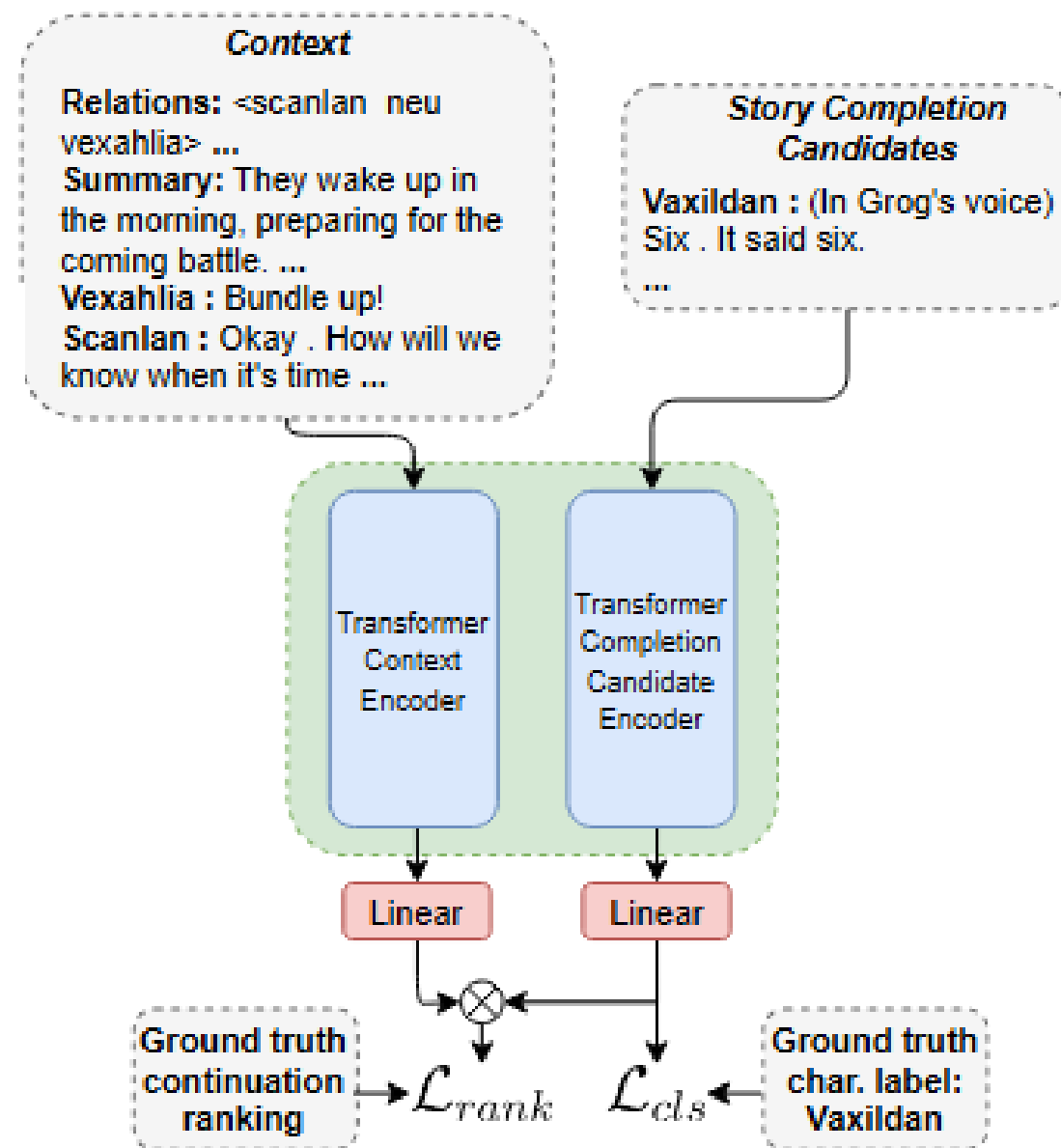
HEART: Human Empathy and Narrative Taxonomy



Relationships

Relations	{ Scanlan, neutral, Vexahlia }, { Keyleth, positive, Scanlan }, { Grog, negative, Vexahlia }, { Scanlan, positive, Vaxildan } ...
Summary	They wake up in the morning, preparing for the coming battle. Scanlan turns them all into Ravenites with light clothing. The sleet storm is starting. ...
Vexahlia:	Bundle up!
Scanlan:	Okay. How will we know when it's time for me to release? We have to wait for Tooma to go report.
Vexahlia:	Is Vorugal back? He's back.
Scanlan:	I assume.
Vexahlia:	<u>Do we see Larkin around?</u>
DM:	<u>No, you do not see Larkin around.</u>
Scanlan:	Vax , do you want to go look?
Vaxildan:	For Larkin?
	No Larkin. <i>I attempt to see see if Tooma is coming.</i> I don't want to release this thing before Tooma is there reporting to Vorugal.
Vaxildan:	(Grog voice) Six. It said six.

Table 1: A sample from CRD3 extended, showing: pair-wise character relationships; historical context via the summary; and current character interactions in the form of dialogue, *first-person* (green), and *second-person* (blue) narration. DM refers to the Dungeon Master who provides arbitration and additional context to players.



Relationships

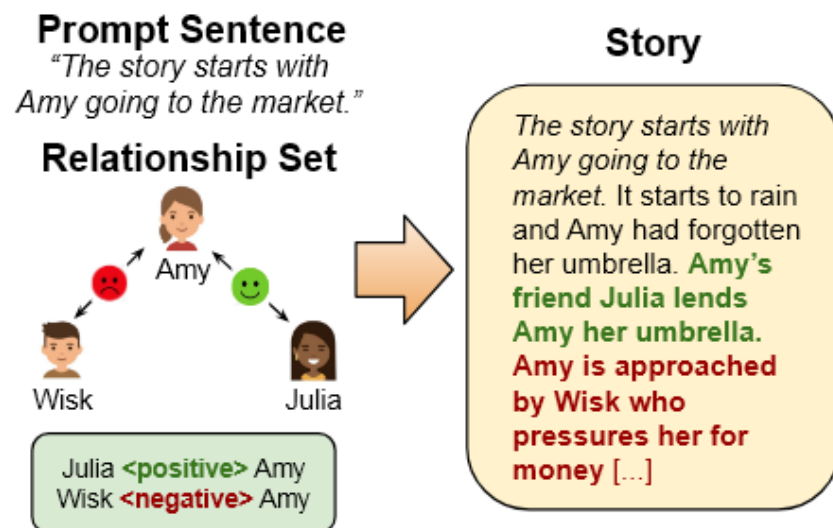


Figure 1: Example of relationship-driven story generation task: given a set of relationships and a prompt sentence, the goal is to generate a story continuing the prompt sentence and reflecting the input relationships. **Positive** and **negative** relationships are highlighted.

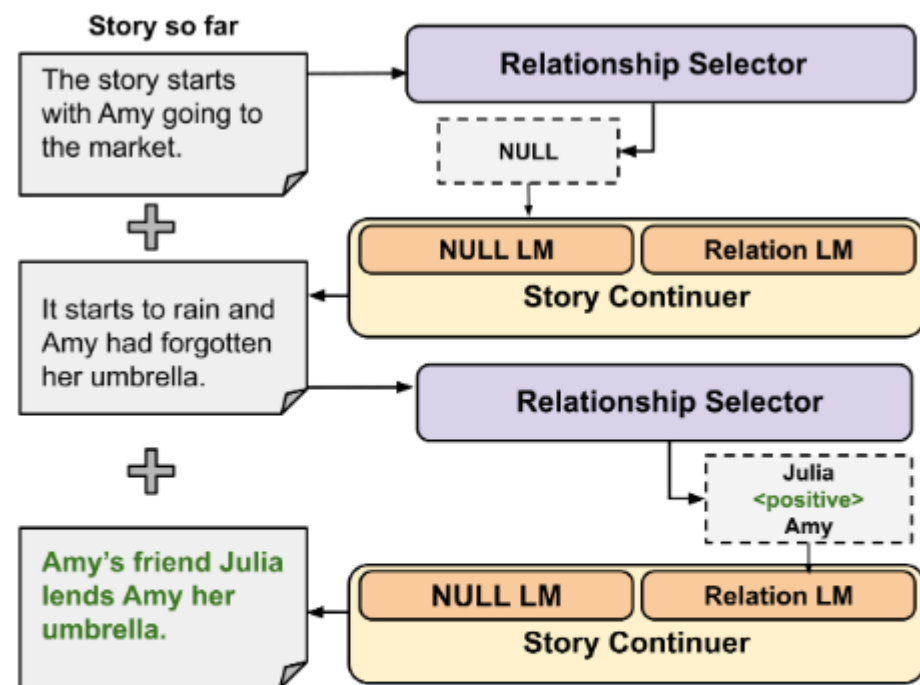


Figure 2: Proposed model RELIST illustrated. RELIST has two components, the relationship selector and the story continuer, which jointly generate the story.

Relationships

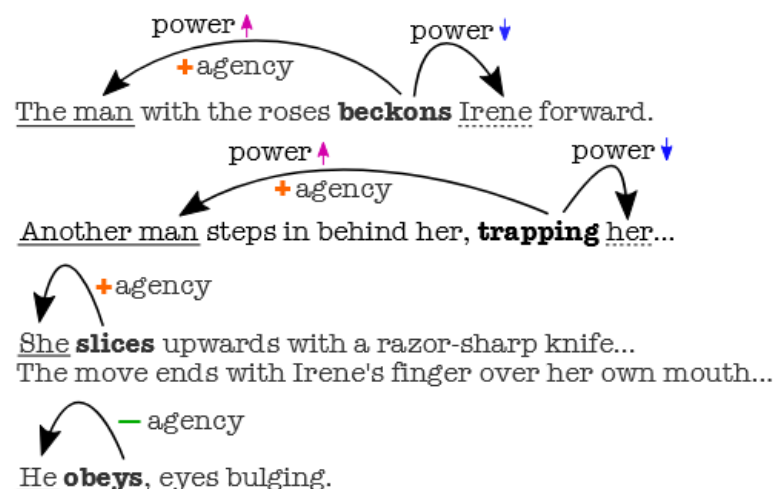
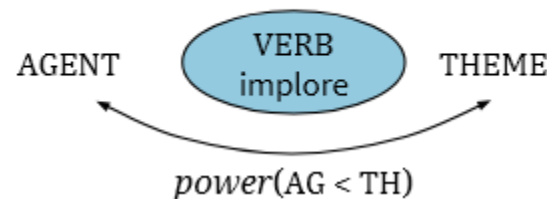
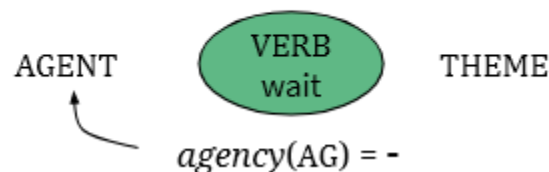


Figure 1: An excerpt from a box-office hit, *Sherlock Holmes* (2009). **Bolded** words are the predicates, solid underlined phrases are the agent of the verb, and dash underlined words are the theme. The full example with additional nuanced discussion is available in Figure 6 in the appendix.

He **implored** the tribunal to show mercy.



The princess **waited** for her prince.



Formal Definition

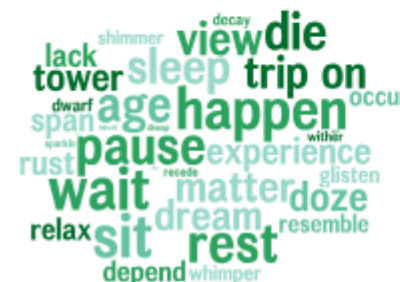
$power(AG < TH)$

$power(AG > TH)$



$agency(AG) = -$

$agency(AG) = +$



Labels

Relationships

```
Megan PROPOSE friend_have_lunch Meredith
Lester PROPOSE friend_chat Robert
Suzette PROPOSE friend_chat Silvy
Betty PROPOSE friend_weekend_out Clark
Meredith PROPOSE mate_watch_tv Lester
Clark REJECT-PROPOSAL friend_weekend_out Betty
Lester REJECT-PROPOSAL mate_watch_tv Meredith
Meredith ACCEPT-PROPOSAL friend_have_lunch Megan
Lester affinity with Meredith 87
Violet PROPOSE friend_chat Megan
Clark affinity with Betty 67
Robert REJECT-PROPOSAL friend_chat Lester
Meredith affinity with Megan 72
Silvy ACCEPT-PROPOSAL friend_chat Suzette
Robert affinity with Lester 72
Betty affinity with Clark 50
(...)
```

```
PLOT-PROJECTION 0
  ProposeActivity {activity=friend_weekend_out, proposee=Clark, proposer=Betty}
PLOT-PROJECTION 1
  ProposedActivityAccepted {activity=friend_weekend_out, proposee=Clark, proposer=Betty}
  AffinityChange {triggerer=Clark, perceiver=Betty, impact=76}
  AffinityChange {triggerer=Betty, perceiver=Clark, impact=51-->54}
PLOT-PROJECTION 2
  ProposeActivity {activity=mate_go_to_cinema, proposee=Mary, proposer=Clark}
PLOT-PROJECTION 3
  ProposedActivityRejected {activity=mate_go_to_cinema, proposee=Mary, proposer=Clark}
  AffinityChange {triggerer=Mary, perceiver=Clark, impact=95}
  AffinityChange {triggerer=Clark, perceiver=Mary, impact=84}
  (...)
```

Knowledge Check

1. Do you think a system can be made that encompasses all of these attributes using today's technology?
2. How would you start making a system like this? (e.g., Would it be LLM-based? Simulation-based? Planning-based?)
3. Would you need all of these to make a “good” story?
4. What about a “realistic” one?

