Proceedings of the Nineteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2023)

Language Model-Based Player Goal Recognition in Open-World Digital Games

Yeo Jin Kim, Alex Goslen, Jonathan Rowe, Bradford Mott, James Lester

North Carolina State University, Raleigh, NC, USA
{ykim32, amgoslen, jprowe, bwmott, lester}@ncsu.edu

Abstract

Devising models that reliably recognize player goals is a key challenge in creating player-adaptive games. Player goal recognition is the task of automatically recognizing the intent of a player from a sequence of observed player actions in a game environment. In open-world digital games, players often undertake suboptimal and varied sequences of actions to achieve goals, and the high degree of freedom afforded to players makes it challenging to identify sequential patterns that lead toward specific goals. To address these issues, we present a player goal recognition framework that utilizes a fine-tuned T5 language model, which incorporates our novel attention mechanism called Temporal Contrary Attention (TCA). The T5 language model enables the framework to exploit correlations between observations through non-sequential self-attention within input sequences, while TCA enables the framework to learn to eliminate goal hypotheses by considering counterevidence within a temporal window. We evaluate our approach using game trace data collected from 144 players' interactions with an open-world educational game. Specifically, we investigate the predictive capacity of our approach to recognize player goals as well as player plans represented as abstract actions. Results show that our approach outperforms non-linguistic machine learning approaches as well as T5 without TCA. We discuss the implications of these findings for the design and development of player goal recognition models to create player-adaptive games.

Introduction

Open-world digital games enable players to immerse themselves in captivating storyworlds featuring non-linear gameplay that prioritizes player agency, while aiming to foster engagement and enhance replayability (Kirginas and Gouscos 2017; Aung et al. 2019). In these environments, players have considerable freedom in choosing what goals to pursue and devising plans to accomplish those goals. Moreover, in these games, players engage in a wide range of actions, many of which are suboptimal, in pursuit of their goals, while being afforded significant flexibility in the exact sequence of actions needed to achieve their objectives. These factors present key challenges for accurately recognizing player goals in these environments.

Player goal recognition refers to the ability of an agent to recognize the intent of players given a sequence of observations. Robust goal recognition that dynamically recognizes and understands the goals of players holds significant potential to create player-adaptive games that tailor gaming experiences to better match players' interest and needs, resulting in more personalized and engaging experiences (Yannakakis et al. 2013; Zhu and Ontanon 2020). While humans tend to perform goal recognition unconsciously through observation, it remains challenging for machines to acquire such ability (Duhamel, Maynard, and Kabanza 2020).

In the field of artificial intelligence, there are two major streams of research on goal recognition: *planning-based* and *data-driven* approaches. Planning-based approaches considers goal recognition as an inverse planning problem, while data-driven approaches use machine learning to recognize goals directly from observations. Planning-based goal recognition has generally been studied using theoretical problems and datasets (Schmidt et al. 1978; Mirsky et al. 2016; Treger and Kaminka 2022; Baker, Tenenbaum, and Saxe 2007; Ramırez and Geffner 2011; Pereira, Pereira, and Meneguzzi 2019), while data-driven goal recognition has primarily been used in simulation and game environments, such as an action-adventure game (Gold 2010), PDDLGym (Amado, Mirsky, and Meneguzzi 2022), and CRYSTAL ISLAND (Mott, Lee, and Lester 2006; Ha et al. 2011; Min et al. 2014, 2016; Goslen et al. 2022a,b).

An important aspect of player goal recognition in open-world digital games is that players often create suboptimal plans. In planning-based goal recognition, a key assumption is optimality, which assumes that the actor is rational and tends to undertake an optimal plan to achieve a goal. However, humans are not always optimal (Treger and Kaminka 2022), and thus we should, in general, assume actions are suboptimal in environments involving human actors. Particularly, in open-world digital games, optimality is not always a valid premise, because players have incomplete knowledge of the gameworld and undertake inefficient paths toward accomplishing goals, while exploring and learning about the environment. While such suboptimal behaviors are permitted and even encouraged in open-world games, this poses significant challenges for goal recognition because suboptimal behaviors are common across different goals, making them less distinguishable. This tendency is further exacer-

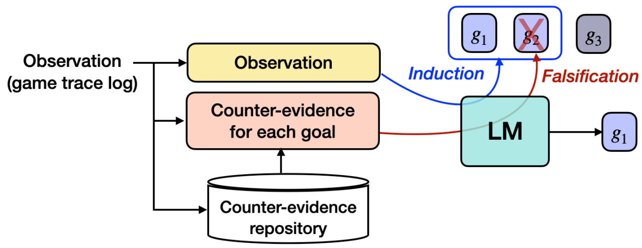

Figure 1: Using counterevidence for goal recognition.

bated when the player is pursuing multiple goals at the same time, which is common in open-world environments.

Another key aspect of player goal recognition in open-world games is its *non-sequentiality*, meaning that a goal is achieved when several conditions are met, and there may or may not be a clear order among these conditions or actions. This is due in part to suboptimality, where temporal patterns are diluted as individual players not only act in a suboptimal order, but also take unrelated or repetitive actions. Thus, under the non-sequentiality assumption, it can be more effective to examine the players' action sequence as a whole and to check whether the requirements constituting a specific goal are met, rather than to find fixed sequences of temporal patterns.

In this work, we investigate the capabilities of language models for player goal recognition. We use a Transformer-based language model, T5 (Raffel et al. 2020), and compare it to six non-linguistic machine learning approaches. While language model-based goal recognition studies are rare, adopting language models to goal recognition problems has three advantages: 1) using game trace logs as text input minimizes feature engineering cost and information loss, 2) the language model can learn *linguistic correlation* among observations while existing non-linguistic machine learning approaches often lose such information, and 3) its *non-sequential self-attention mechanism* offers promise for dealing with issues associated with non-sequentiality and suboptimality.

Furthermore, we introduce a novel attention mechanism we call *Temporal Contrary Attention (TCA)* in this work. The self-attention mechanism utilized by T5 differentially weighs the significance of each element of the input sequence relative to the target label so that the model pays more attention to small but important parts of the input. Unfortunately, this can result in overlooking the occurrence of rare events that might eliminate false hypotheses. For example, consider the scenario illustrated in Figure 1 where most of the observations point to the first goal, g_1, or the second goal, g_2, as probable goal hypotheses based on *induction*. When g_2 has a higher probability given observations, the model will recognize the goal as g_2, which might be a statistical prejudice in this case. However, if the model obtains counterevidence against g_2 (i.e., observations that are unlikely to occur with g_2), the model will learn to exclude the corresponding goal hypothesis. By examining the relationship between true and false goal hypotheses simultaneously, the model can learn the ability to infer goals through the use of *falsification*, which is an inference technique that denies a proposition by presenting counterexamples. To facilitate the learning of falsification by the language model, TCA applies self-attention to the least likely observations that are pertinent to false hypotheses.

To evaluate our work, we use data from an open-world educational game, CRYSTAL ISLAND, used in prior research on goal recognition (Goslen et al. 2022a) and employ T5 (Raffel et al. 2020) as our base language model. The results show that the T5-based models outperform non-linguistic machine learning approaches, and that T5 with TCA further improves the goal recognition performance.

Related Work

This work lies at the intersection of goal recognition, language models, and reasoning with language models.

Goal Recognition

Plan, activity, and goal recognition have been the subject of extensive work (Sukthankar et al. 2014). As noted above, there are two major approaches to goal recognition: one views goal recognition as an inverse planning problem, and the other leverages data-driven techniques to recognize goals directly from observations.

Planning-based goal recognition includes traditional plan library-based goal recognition and recent studies using automatic planning techniques with a domain model, often called goal recognition as planning or model-based goal recognition. Plan-library based goal recognition encodes all possible observable plans, and a goal is inferred by searching a matched plan (Schmidt et al. 1978; Kautz and Allen. 1986; Charniak and Goldman 1993; Mirsky et al. 2016; Treger and Kaminka 2022). Recently, Rabkina et al. (2022) proposed a cognitive model-based goal recognition framework using the analogy of similar cases retrieved from a plan-library and evaluated it using MineCraft and Monroe. A key issue with these approaches is that as the problem space grows, it becomes increasingly difficult to generate and retrieve all possible plans. To overcome this limitation, model-based goal recognition has emerged, which dynamically generates optimal plans with observations using a planning algorithm in a known domain model, and then estimates a goal with the highest likelihood among goal hypotheses given generated optimal plans (Baker, Tenenbaum, and Saxe 2007; Ramırez and Geffner 2011; Pereira, Pereira, and Meneguzzi 2019; Shvo and McIlraith 2020; de A. Santos et al. 2021; Masters and Vered 2021). However, in open-world digital games, these planning-based goal recognition approaches are often less practical for two reasons: 1) the optimality assumption is invalid for human players, and 2) the expansive nature of the game world presents challenges in developing computational domain models incorporating all the possible states and player actions.

Data-driven goal recognition leverages machine learning techniques, which is suitable for open-world digital games where players' actions are suboptimal and the domain model is unknown. In CRYSTAL ISLAND, Mott, Lee, and Lester

(2006) proposed probabilistic goal recognition models using n-gram models and Bayesian networks, which were followed by explorations of several more advanced methods to improve the recognition accuracy of the models, including Markov Logic Networks (Ha et al. 2011) and Autoencoders (Min et al. 2014). Due to the sequential nature in recognizing goals from gameplay trace data, Long Short-Term Memory (LSTM) networks were found to increase predictive performance (Min et al. 2016). Goal recognition in CRYSTAL ISLAND was further explored using player-created plans as labels, with LSTMs still outperforming other machine learning models (Goslen et al. 2022a,b).

In other simulation or game environments, Gold (2010) utilized Input-Output Hidden Markov Models (IOHMM) to recognize goals from low-level actions in a top-down action adventure game, while Nguyen et al. (2011) leveraged Markov decision processes to recognize the goals of human players in a collaborative maze-style game. Recently, Pereira, Pereira, and Meneguzzi (2019) combined deep learning and planning techniques to address incomplete domain model issues with a planning dataset encoded using STRIP and PDDL, and Duhamel, Maynard, and Kabanza (2020) combined transfer learning and few-shot learning to infer goals in a navigation problem within synthetic grid-world maps from StarCraft. Amado, Mirsky, and Meneguzzi (2022) also used deep learning to reduce the need for domain knowledge in PDDLGym. Although a diverse set of machine learning and deep learning methods have been investigated for goal recognition, language models have been under-explored despite their potential.

Language Models

In recent years, language models have demonstrated significant success across a wide array of natural language processing tasks (Zhao et al. 2023; Mialon et al. 2023), while enabling the creation of impressive language-related tools such as Copilot (Chen et al. 2021) and ChatGPT evolved from GPT-3 (Brown et al. 2020). Many of the recent language models, such as BART (Lewis et al. 2020), T5 (Raffel et al. 2020), LaMDA (Thoppilan et al. 2022), are based on the Transformer architecture (Vaswani et al. 2017), which utilize an encoder-decoder approach where the encoder abstracts an input sequence and the decoder generates a desired output sequence from the abstracted embedding. A key component of the Transformer architecture is a self-attention mechanism, with which the model is trained to pay attention to input tokens that are more important with respect to outputs. To the best of our knowledge, transformer-based language models have not been used for inferring goals in open-world digital games. We believe that language model-based approaches to goal recognition hold significant promise.

Reasoning in Language Models

Reasoning is the ability to make inferences using evidence and logic to reach a conclusion. Formal reasoning includes deduction, induction, and abduction, while informal reasoning is a less structured approach relying on intuition, experience, and common sense to draw conclusions. Recent research shows that a certain level of reasoning ability emerges in large language models, which generally contain 100+ billion network parameters, and it is rarely inherent in small-scale language models (Wei et al. 2022; Suzgun et al. 2022). According to Huang and Chang (2022), there are three main ways to implement reasoning with language models: supervised fine-tuning (Rajani et al. 2019; Hendrycks et al. 2021; Nye et al. 2021), prompt engineering (Wei et al. 2022; Fu et al. 2023; Wang et al. 2023; Zhou et al. 2023), and reasoning-enhanced training and prompting using a hybrid approach (Lewkowycz et al. 2022; Chung et al. 2022; Anil et al. 2022). Supervised fine-tuning and hybrid approaches to reasoning are applicable to both small and large-scale language models, while prompt engineering for reasoning is only possible in large language models because it depends on the model's innate reasoning power. Prompt engineering helps with reasoning, but it does not enhance the reasoning ability of the language model itself because it does not update the network parameters. On the other hand, hybrid approaches can improve the reasoning ability of the language model by updating network parameters, so that the language model can solve more complex and domain specific reasoning problems. We take a hybrid approach where we fine-tune T5 (Raffel et al. 2020) with prompting engineering for falsification, in which we actively append counterevidence for each goal.

Methods

Our approach is a model-free data-driven goal recognition approach using a fine-tuned language model with prompt engineering for reasoning. In model-free goal recognition, a domain model is unknown, and states are latent. Thus, the goal recognition model must be learned from data and given set of goal hypotheses without a domain model. Moreover, in many game environments, observations often have long, dynamic trajectories, and the exact timing of goal recognition cannot be specified in advance. Thus, we leverage recent observation sequences at any point in time as input rather than having observations from the beginning of the game up to a specific time step.

Definition 1. A model-free goal recognition problem is defined as $\Pi_{\mathcal{G}}^{\mathcal{O}} = (\mathcal{M}, \mathcal{G}, \mathcal{O})$ where $\mathcal{M}$ is a recognition model, $\mathcal{G} = \{g_1, g_2, ..., g_m\}$ is a set of goal hypotheses, and $\mathcal{O}$ is a sequence of observations, while $\mathcal{O}_i = (o_{i-l+1}, ..., o_{i-1}, o_i)$ is an l-length observation sequence ending at time step i in $\mathcal{O}$. Since the player can pursue multiple goals at the same time in our task, the solution to the goal recognition problem is the set of target goals $\mathcal{G}_i^* \subseteq \mathcal{G}$ being pursued at time step i in $\mathcal{O}_i$.

Base Language Model

We leverage T5 (Raffel et al. 2020) as our base language model to recognize goals. T5 is a Transformer-based language framework where all natural language processing (NLP) tasks are reformulated into a unified text-to-text-format, i.e., the input and output are always text strings. The architecture of T5 builds upon the original Transformer architecture (Vaswani et al. 2017) using three main modifications: removing the Layer Norm bias, placing the layer

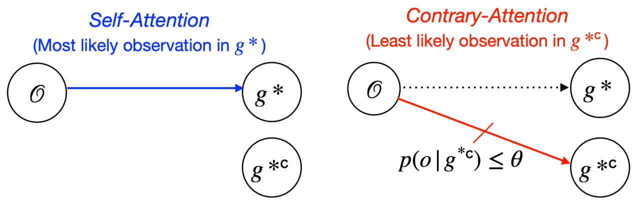

Figure 2: Self-attention vs. Contrary-attention.

normalization outside the residual path, and using a different position embedding scheme. T5 models were pretrained using the Colossal Clean Crawled Corpus (C4), which is a pre-processed English language corpus that is approximately 700GB in size. Among the several sizes of available T5 models (Raffel et al. 2020), we fine-tune T5-small (600MB in size with 60M parameters) for our goal recognition models using engineered prompts for reasoning. We choose the small version of T5 models taking into account effective model training and deployment of the model to support real-time goal recognition in the future. For simplicity, henceforth in the paper T5-small is denoted as T5.

Temporal Contrary Attention (TCA)

In general, when a language model is fine-tuned for goal recognition, the $<$*input*, *output*$>$ format is expressed as $<$*observation sequence*, *true goal labels*$>$. Then the self-attention mechanism learns to pay attention to the correlations between the observation sequence, $\mathcal{O}$, and the true goal set, $\mathcal{G}^*$, during training, but it tends to ignore the correlation of the input with false goals, i.e., the complement set of true goals, $\mathcal{G}^{*\mathsf{c}}$, which are grounds of falsification. Figure 2 illustrates that self-attention (Left) focuses on learning how *strongly* observations in the input sequence are correlated to a target goal set, $\mathcal{G}^*$, while contrary attention (Right) learns how *weakly* each observation is connected to the complement set of target goals, $\mathcal{G}{*}^{\mathsf{c}}$. To enable the model to learn how to eliminate false goal hypotheses directly from observations, we first need to collect observations which are unlikely to occur with a specific goal hypothesis, which we refer to as *least likely observations*.

Definition 2: Least Likely Observation (LLO). When an observation o occurs with a likelihood $p(o|g) \leq \theta$ for a goal $g \in \mathcal{G}$, o is a least likely observation (LLO) for g, denoted as $o \in \mathcal{L}_g$, where θ is a very small probability near zero.

Note that when $\theta = 0$, a LLO might be interpreted as a counterexample for deduction in formal logic, but since LLOs are being collected from training data, LLOs in test data can still have a small, positive probability of occurring for a goal ($\theta > 0$). This suggests that we cannot reject a particular goal hypothesis just because we have observed a few LLOs. Instead, the model needs to consider both the LLOs and the original observation sequence comprehensively to decide whether it should exclude a goal hypothesis or not.

Contrary Attention (CA). We refer to the inference approach that focuses on the relationship between observations and false hypotheses as *contrary attention (CA)* in contrast to self-attention, which focuses on the correlation between the observation sequence and true hypotheses. To add contrary attention to our model, the $<$*input*, *output*$>$ representation is replaced with $<$*input*, *counterevidence*, *output*$>$ in our prompt engineering. In CA, counterevidence includes the LLOs for each goal.

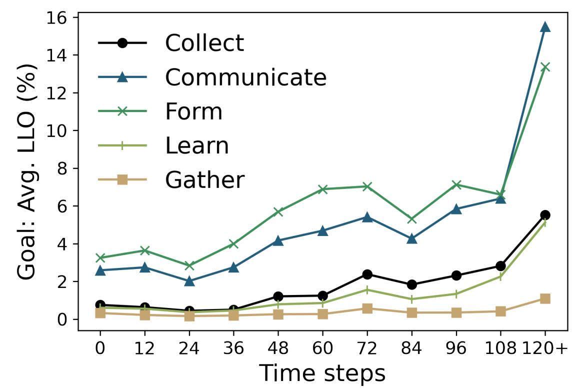

Figure 3: Example of the average percent of least likely observations (LLOs) among observations over time toward completing goals in our data set.

One issue with CA is that as the number of suboptimal behaviors increases, the size of the set of LLOs decreases and the potential benefit of the approach decreases overall. In general, the closer a player gets to achieving a goal, the more optimized their actions are towards the goal (i.e., a smaller number of distinct observations with less suboptimal actions), which means that the ratio of LLOs among observations increases over time in goal trajectories, as shown in Figure 3. Since the types of suboptimal actions can vary during different periods in goal trajectories, setting a temporal window for LLO analysis can increase the benefits of the approach by separating the interference of suboptimal actions among different time periods. The smaller the temporal window, the more selectively the LLOs can be defined; however, the size of temporal window cannot infinitely decrease, because it can overfit the training data. Achieving generalizability requires a temporal window large enough to avoid overfitting, but small enough to selectively detect changes in observations over time, which implies that the optimal temporal window size depends on the data. In our work, to find the optimal temporal window size during training, a grid search was performed with a sliding temporal window size, t_w, as a hyperparameter; we defined a temporal window $[t_s, t_e]$ with a start time ($t_s = i - t_w$) and an end time ($t_e = i + t_w$) based on the current time step i. We name this temporal window-based LLO as *Temporal LLO*.

Definition 3: Temporal LLO (TLLO). When an observation o occurs with a likelihood $p(o|g) \leq \theta$ for a goal $g \in \mathcal{G}$ given a temporal window $[t_s, t_e]$, o is a temporal least likely observation for g, denoted $o \in \mathcal{L}_{g,[t_s,t_e]}$. LLO is a special case of TLLO with the temporal window covering the entire sequence.

Algorithm 1 shows the steps for TLLO analysis for the

Algorithm 1: Temporal least likely observation (TLLO)

Input: goal hypotheses $\mathcal{G}$, observations $\mathcal{O}$
Parameter: start time t_s, end time t_e
Output: $\mathcal{O}^{\mathcal{L}}_{\mathcal{G}}$ = TLLO sequence for $g \in \mathcal{G}$

for all $g \in \mathcal{G}$ **do**
  $\mathcal{L}_{g,[t_s,t_e]} \leftarrow \textsc{CollectTLLO}(\mathcal{G}, \mathcal{O}_{[t_s,t_e]})$
end for
for all $g \in \mathcal{G}$ **do**
  for all $o \in \mathcal{O}_{[t_s,t_e]}$ **do**
    if $o \in \mathcal{L}_{g,[t_s,t_e]}$ **then**
      $\mathcal{O}^{\mathcal{L}}_{g} \leftarrow$ add o
    end if
  end for
end for
return $\mathcal{O}^{\mathcal{L}}_{\mathcal{G}}$

set of goals $\mathcal{G}$, given $\mathcal{O}$ with a temporal window $[t_s, t_e]$. First, for each goal g, it collects LLOs from $\mathcal{O}_{[t_s,t_e]}$, which is all the observation sequences within a temporal window $[t_s, t_e]$, and stores the TLLOs to a counterevidence repository $\mathcal{L}_{g,[t_s,t_e]}$. Second, for each goal g and each observation o, given the temporal window, if o belongs to $\mathcal{L}_{g,[t_s,t_e]}$, it aggregates o as counterevidence against g and stores them in the return sequence $\mathcal{O}^{\mathcal{L}}_{\mathcal{G}}$. Intuitively, the longer a TLLO sequence is for a goal, the higher chance the goal is excluded.

Temporal Contrary Attention (TCA). We name our inference approach that focuses on the relationship between temporal window-based counterevidence and false hypotheses as *temporal contrary attention (TCA)*. TCA can be implemented by combining the observation sequence with TLLO sequences as input to T5 using <*input*, *temporal counterevidence*, *output*> as the representation for prompt engineering where the temporal counterevidence consists of TLLOs. Then the model is fine-tuned with the training data containing the engineered prompt to enable the model to learn the reasoning technique of falsification.

T5 prompt engineering. Figure 4 shows examples of the input and output formats we use for T5, CA-T5, and TCA-T5. While T5 has an only the observation sequence $[o_{i-l+1,\ldots,o_{i-1},o_i}]$ as input, CA-T5 has a LLO sequence for each goal without a temporal window, and TCA-T5 has a TLLO sequence with a temporal window $[t_s, t_e]$ as an additional input. In CA-T5 and TCA-T5, the model focuses on the correlation between observations and true labels as a primary attention function and the correlation between LLOs/TLLOs and false labels as a secondary CA/TCA function.

Experiments

We briefly introduce the game environment and dataset and then describe the experimental setting for our evaluation.

Game Environment

To validate our goal recognition approach, we utilized the dataset and labels provided by prior work using CRYSTAL ISLAND as a testbed (Goslen et al. 2022a). The dataset

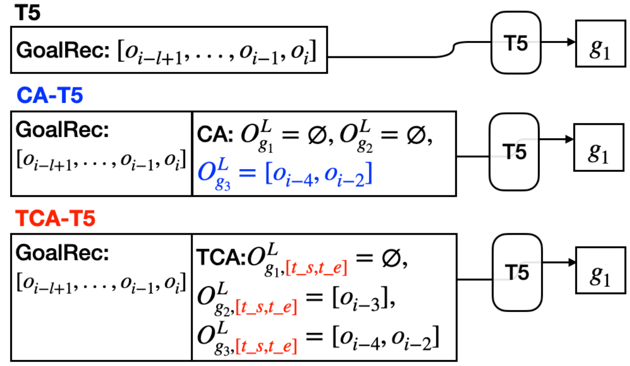

Figure 4: Examples of prompt engineering for goal reasoning in T5, CA-T5, and TCA-T5.

Figure 5: The CRYSTAL ISLAND game-based learning environment.

consists of 144 eighth grade students' interactions with the CRYSTAL ISLAND game-based learning environment, which is an open-world digital game designed for middle school science education (Figure 5). In the game, players investigate a mysterious outbreak on a remote island research station by exploring different locations, conversing with non-playable characters (NPCs), reading books and posters, and testing items in a virtual laboratory. The game offers an open-world experience from a first-person perspective.

During gameplay, players were prompted to specify goals and create plans for achieving the goals using a block-based visual interface (Figure 6). Goals are represented using a high-level goal clamp in the planning tool, while plans consist of a series of nested abstract actions. Players could voluntarily access the interface throughout their gameplay. The game automatically recorded and stored all actions and states that occurred during gameplay, including the use of the planning tool, in game trace logs. Prior work derived goal and plan recognition labels from student interactions with the interface using the trace logs along with low-level in-game actions serving as observations for the machine learning models.

Assumptions: Table 1 summarizes the assumptions of our goal recognition problem in CRYSTAL ISLAND, following the assumption criteria of Masters and Vered (2021). In

Category	Environment		Observations		Agents		Goals	
Assumptions	Continuous	×	Missing	×	Non-Keyhole	×	Non-Equal Priors	✓
	Online	×	Noisy	×	Different Model	✓	Multi-Goals	✓
	Dynamic	×	Fluents/States	✓	Suboptimal	✓	Sub-Goals	×
	Stochastic	×			Fallible Observer	×	Dynamic Goals	✓
	Partial Observability	×						

Table 1: Assumptions for goal recognition in CRYSTAL ISLAND.

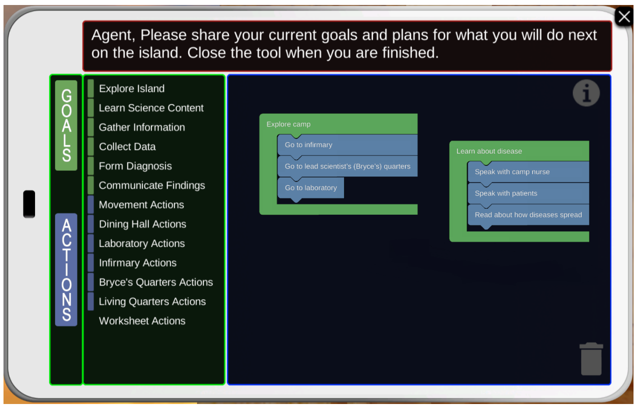

Figure 6: Planning support tool with block-based visual interface.

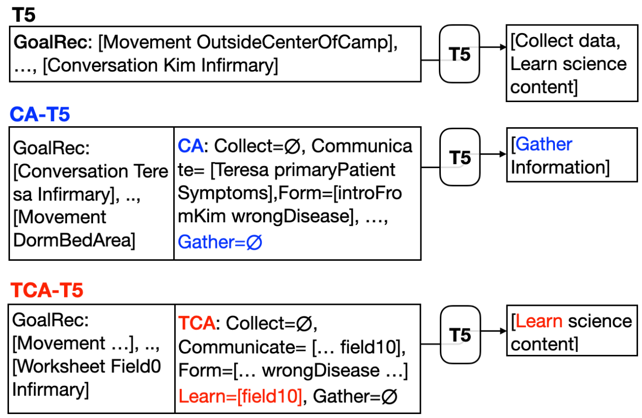

Figure 7: Examples of Temporal Contrary Attention to least likely observations.

CRYSTAL ISLAND, the environment is continuous since the player's behavior is open and infinite, but the continuous actions and states were discretized to high-level concepts and locations in the provided data, and thus it is *discrete*. When interacting with players, the game environment is online, but we utilized *offline* data collected from a prior study (Goslen et al. 2022a). This particular version of game is also *static* and *deterministic* because the environment's rules do not change over time, and there is no randomness in state transition. In observations, there is *no missing value or noise*, while we used *states* like current locations as observations along with students' actions. We hold the keyhole assumption that the player is unaware of the existence of the observer (i.e., the goal recognition model). We also assume that the player and the observer have *different models* because the observer does not know the player behavior model. The player's behaviors are *suboptimal*, while the observer is *not fallible* in that all observations are recorded in the game trace log and cannot be misinterpreted. Lastly, goals are *non-equal priors* because not all goals need to be achieved to complete the game, and each goal is established *dynamically* by the player. *Multi-goals* are allowed but there are *no sub-goals* in a hierarchical way, rather they are unordered and some are optional.

Data: The study data was collected from 144 eighth-grade students, with 60% of them being female and 40% male. The students played CRYSTAL ISLAND remotely (due to the pandemic) during their science classes over two days. The students were not restricted by any time limit, and on average, they played the game for 94.7 minutes with a standard deviation of 47.7 minutes.

Each player had different interactions with the planning support tool with various event sequence lengths. The player trajectories were segmented to event sequences by each interaction with the planning support tool. The event sequences were constructed cumulatively for action-level prediction. The maximum event sequence length was set to 30, the median of sequence length across players, with zero-padding. The number of event sequences is 385, and the total number of action events is 11,550.

For the baseline methods, the sequences were represented by one-hot encoding vectors, following (Goslen et al. 2022b; Min et al. 2017). For T5-based models, the same sequences were used in text string format to retain the same amount of information as the baselines. Figure 7 shows examples of the input and output formats utilized for T5, CA-T5, and TCA-T5 where each element of the input observation sequence and the output labels are considered as a word token. For CA-T5 and TCA-T5, we add least likely observations (LLOs) for each goal candidate to the original input sequence that helps the model learn falsification. The only difference between CA-T5 and TCA-T5 is whether LLOs or TLLOs are utilized.

Labeling: Generally, labeling of player goals and plans is non-trivial since it requires knowing the players' intention during gameplay. Prior work (Goslen et al. 2022a) encouraged players to externalize their goals and plans through the use of a planning support tool. The planning support tool offered players a choice of 20 goals, categorized into 5 goal classes, and 55 available abstract actions that players could use to create plans for their goals. As with the prior work, goal labels were generated through the goal categories students selected in each planning support tool interaction, and actions were first categorized into 6 abstracted actions and then eventually clustered into 4 classes, using SpaCy word embeddings (Levy and Goldberg 2014; Srinivasa-Desikan 2018) and k-means clustering. Both goal and plan recognition are formalized as a multi-label classification problem because players are not assumed to have a single goal at a time and plans usually contained multiple actions for a selected goal. Goals and plans that had already been completed were removed from the label set to make the labels closer to the ground truth. The final distribution of goals in the dataset is: Collect Data (22%), Communicate Findings (5%), Form Diagnosis (6%), Learn Science Content (22%), and Gather Information (46%). The plan labels are: Read Science Content (10%), Explore (28%), Gather and Scan Items (34%), Speak with Characters (28%).

Experiment Setting

For evaluation, we compare six non-linguistic machine learning methods as our baselines with three proposed language model-based methods as follows:

- **Baselines**: Random Forest (RF), Multi-Layer Perceptron (MLP), Support Vector Machine (SVM), Gaussian Naive Bayes (GNB), Logistic Regression (LR), Long-Short Term Memory (LSTM)
- **Proposed**: T5, Contrary Attention with T5 (CA-T5), Temporal Contrary Attention with T5 (TCA-T5)

For CA, we first construct a counterevidence repository, L_g, for each goal g, using the training data and utilized L_g to generate LLO sequences for each input for both the training and test data in cross validation. In this work, we set $\theta = 0$ in $p(o|g) \leq \theta$ to define LLO. For TCA, the counterevidence sequence generation is similar to CA except that it builds a temporal counterevidence repository, $L_{g,[t_s,t_e]}$, and generate TLLO sequences for each input. To define TLLO, we set $\theta = 0$ in $p(o|g) \leq \theta$ as in CA. Note that LLOs and TLLOs are extracted from O_i but not from any future observations, when they match the counterevidence in L_g and $L_{g,[t_s,t_e]}$, respectively.

Metrics: Since the data is highly imbalanced, we used macro F1-score as our overall performance evaluation metric. When the recognition result includes a true label among multiple labels for a data instance, the corresponding recognition was considered as a correct inference. We made this assumption, since the player may be pursuing a single goal at any time step, even when the player established multiple goals in one session. Also, we examine both the absolute and relative increase of macro-F1 across classes to assess performance improvement compared to the best baseline.

Hyperparameters and systems: In the T5-based models, we searched the learning rate in [0.0002, 0.0003, 0.0005] with fp16. We fixed other parameters of the language model: batch_size=4, input max_length=1024, weight_decay=0.01, warmup=1000. For the baseline hyperparameters, see Appendix. We did 5-nested cross validation to search for an optimal hyperparameter during evaluation on a system with a GPU with 8GB memory (NVIDIA GeForce RTX 2060 super) and a CPU with 16 cores (AMD EPYC 7302P). The 5-nested cross validation for each goal/-plan recognition task took about GNB (0.2 min), RF (11 min), LSTM (25 min), MLP (1.4 hours), LR (3.3 hours), SVM (3.9 hours), and T5-based models (8 hours).

Results

Table 2 shows the results for goal recognition for each of the baseline methods as well as the language model-based methods. Column 3-7 show the F1-score for each of the individual goal classes, and the Overall column shows the overall goal recognition performance with macro-F1 score across the 5 goal classes. The absolute and relative increases in the final column was calculated based on the macro-F1 of the best baseline for goal recognition, LSTM. Among the baselines, LSTM (45.2%) outperformed the other baselines, followed by LR (39.5%) and GNB (38.1%). The T5-based models outperformed all the non-linguistic baselines. Among the T5-based models, TCA-T5 achieved the best goal recognition rate with 56.5% of macro-F1; the absolute and relative increase from the best baseline, LSTM, is 11.3% and 25.0%, respectively.

Table 3 shows the results for plan recognition. The format of table is similar to the goal recognition table, while the absolute and relative increase was calculated based on the macro-F1 of the best baseline for plan recognition, GNB. We observe that plan recognition results are slightly different from goal recognition results in the baseline methods. Among the baselines, GNB (45.3%) is the best, followed by LSTM (42.8%) and LR (40.9%). T5-based models outperformed all the non-linguistic baselines again. Among the T5-based models, TCA-T5 produced the best plan recognition rate with 55.4% of macro-F1; the absolute and relative increase from the best baseline, GNB, is 10.1% and 22.3%, respectively. In plan recognition, it is noted that in the T5-based models, CA does not help to improve the macro-F1, while TCA shows less improvement, compared to goal recognition. We further discuss this result in the Discussion section.

Temporal window size: Figure 8 shows the goal recognition rate (left) and the plan recognition rate (right) by increasing the temporal window size in TCA-T5. We found that

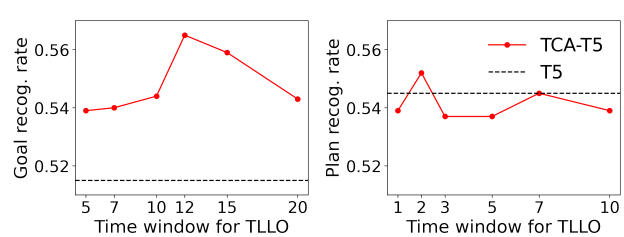

Figure 8: Recognition rate (macro-F1) for TLLO with different temporal window sizes.

		Collect data	Communicate Findings	Form diagnosis	Learn science content	Gather info.	Overall [macro-F1]	Absolute / Relative
	N-distribution	(22%)	(5%)	(6%)	(22%)	(45%)	(100%)	Increase (%)
Baseline	RF	0.1	0.0	0.0	0.0	**67.3**	13.5	-31.7 / -70.1
	MLP	22.4	14.4	30.4	14.4	62.0	28.7	-16.5 / -36.5
	SVM	22.6	24.5	37.0	21.6	66.5	34.5	-10.7 / -23.7
	GNB	**45.9**	14.7	25.7	**41.5**	62.7	38.1	-7.1 / -15.7
	LR	31.7	29.5	45.9	26.4	64.1	39.5	-5.7 / -12.6
	LSTM	36.2	**48.0**	**47.0**	33.8	60.4	**45.2**	0.0 / 0.0
Ours	T5	55.4	51.6	49.1	46.1	**55.4**	51.5	6.3 / 13.9
	CA-T5	54.7	**51.8**	50.8	50.0	55.0	52.5	7.3 / 16.2
	TCA-T5	**55.3**	50.2	**74.3**	**47.5**	55.3	***56.5**	**11.3 / 25.0**

Table 2: Results for the goal recognition task.

		Read science content	Explore	Gather & Scan items	Speak with characters	Overall [macro-F1]	Absolute / Relative
	N-distribution	(10%)	(28%)	(34%)	(28%)	(100%)	Increase (%)
Baseline	RF	32.2	29.9	12.7	7.0	14.1	-31.2 / -68.9
	MLP	33.4	36.1	37.6	30.1	34.3	-11.0 / -24.3
	SVM	28.7	31.8	35.6	28.5	31.1	-11.2 / -24.7
	LR	**40.4**	46.0	30.7	**46.5**	40.9	-4.4 / -9.7
	GNB	39.0	**50.6**	46.7	45.0	**45.3**	0.0 / 0.0
	LSTM	31.0	46.2	**53.4**	40.0	42.8	-2.5 / -5.5
Ours	T5	**57.1**	59.2	49.1	52.1	54.5	9.2 / 20.3
	CA-T5	56.7	59.9	50.5	50.9	54.5	9.2 / 20.3
	TCA-T5	55.3	**61.1**	**52.6**	**52.5**	***55.4**	**10.1 / 22.3**

Table 3: Results for the plan recognition task.

for goal recognition the optimal temporal window size was 12-time steps, while for plan recognition 2-time steps were optimal. This finding can be partially explained by goal trajectories have longer timespans than plan trajectories in our work, since a goal trajectory consists of one or more plan trajectories. In this sense, the optimal temporal window size for goal recognition is likely to be longer than the one for plan recognition. Across different temporal window sizes, the goal recognition rates of TCA-T5 are consistently and significantly above the base T5 model, while the plan recognition rates of TCA-T5 are generally lower than T5 except on the optimal temporal window size of 2-time steps.

Discussion

We empirically evaluated our language model-based goal recognition approaches using data from an open-world educational game. Results show that all the language models outperform non-linguistic machine learning models, and our TCA-T5 model achieved the best performance in both goal and plan recognition tasks by encouraging the language model to learn a falsification technique. In this section, we discuss three main topics: labeling, comparison of goal and plan recognition, and generalizability with reasoning.

Labeling. Labeling players' goals and plans in a game is a challenging task for three reasons. First, it is tricky for even the players to clearly know their goals and plans in advance, and players' actions may deviate from pursuing their goals and executing plans while playing. It is questionable to what extent it is acceptable to classify such deviant behaviors as labeled goals and plans. In our task, plans were clustered as higher-level concepts using a clustering method, so gaps and errors coming from the process of clustering could not be overlooked. This is a fundamental issue that arises even if we encourage players to explicitly externalize their goals and plans by providing a planning support tool. Second, allowing multi-labels blurs the boundaries between two or more goals and plans in one trajectory. Although a player may have more than one goal or plan in mind when establishing them, an action at a moment is usually taken for a single goal or plan. In prior work (Goslen et al. 2022a), they removed completed goals and plans from among multi-labels to partially alleviate this problem, but the issue remains while multi-labels are maintained until only a single label remains. As a result, many actions performed for a single goal or plan can become unclear under multi-labels. Additional work could be done to better detect which plans students are actively enacting at a given time, lessening the noise in a goal recognition model. Third, in a real game environment, knowing when to prompt players to plan is difficult. Prompting to use the planning tool encouraged players to update their plans, however there were some cases of players not updating their plans, implying that we did not always have a proper representation of their goal at a given time. Under this circumstance, clustering-based automatic labeling may be an alternative approach, which can serve as a way to evaluate the labels generated from players' plans.

Goal vs. plan recognition. In the experimental results, goal and plan recognition showed different patterns among compared methods. A finding is that TCA-T5 showed a major improvement of 5.0% from the base T5 model for goal recognition, whereas only a minor improvement of 0.9% for plan recognition. In other words, the falsification technique was highly effective for goal recognition but less effective for plan recognition. It is noted that achieving a goal generally spans a longer time than executing a plan. TCA can be viewed as attention to temporal patterns relaxed by a temporal window in that it captures the gradual change in the composition of observations (actions) toward an objective and weighs the possibility of its occurrence. If the observation sequence is too short, it can fail to grasp a clear change in observations and lose its generalizability. It appears that TCA better perceives sufficiently long patterns rather than short ones particularly plagued by suboptimal actions.

This temporal characteristic is also revealed in the baseline methods. All of the non-temporal methods as well as T5 and CA-T5 except SVM showed better performance in plan recognition than goal recognition, while LSTM, a temporal method, dropped by 2.4% in plan recognition compared to goal recognition. That is, in terms of non-temporal methods, plan recognition is an easier problem than goal recognition. This suggests that, at least for this dataset, the length of time required to execute goals and plans characterizes the problems and influences the choice of recognition method.

Generalizability with reasoning. TCA-T5 is implemented as a hybrid approach using a language model that combines prompt engineering and fine-tuning to learn a falsification technique through a collection of counterevidence toward labels. Unlike the usual zero-shot or few-shot learning in large language models (LLMs), our method actively collects evidence beyond adding simple instructions for reasoning to prompts. This evidence can be extracted only from the target data, and thus even LLMs do not contain this information unless they are explicitly trained on this knowledge through fine-tuning. Our hybrid language model learning using active prompting and training for reasoning is a general method, applicable to various text input-based sequential classification problems such as recognition of player emotions, game skill levels, or player types. Since T5 is used as the base language model, multi-task classification using multi-prefixes is also possible; by using different prefixes with the same input, a single model can serve diverse classification problems to support players, thereby improving the overall game experience and player engagement.

In addition, although the language model has the disadvantage of being large and expensive to train compared to other non-linguistic machine learning models, the classification at run-time is real-time. On the other hand, since T5-small, a small-scale language model, is relatively lightweight compared to large language models, it has the advantage of being able to learn on a single machine. Lastly, it is worthy to mention that it is not clear whether TCA-T5 learns to infer falsifications in a human-like way. Rather, if the Temporal LLO sequence of a specific label is long or contains important keywords, it is likely to learn to exclude that label with high probability. There has been a similar debate in large language models where simple reasoning ability has emerged (Huang and Chang 2022). Thus, further analysis and research will be needed to understand the underlying mechanism of reasoning in language models.

Conclusion

In an open-world digital games, goal recognition has an important role in understanding and supporting players. Using language models for goal recognition introduces new possibilities for creating player-adaptive open-world digital gaming experiences. In this work, our contributions are summarized in three points. First, we present a language model-based goal recognition framework that utilizes linguistic correlation among observations and order-independent self-attention. This resulted in improved performance for goal and plan recognition compared to competitive non-linguistic machine learning methods. Second, we present a novel algorithm for Temporal Contrary Attention (TCA) to help the model learn a falsification technique to eliminate hypotheses by considering counterevidence. This approach outperformed the T5 models and achieved the best performance particularly for the goal recognition task. Finally, we analyzed three major design factors in goal recognition in open-world digital games, including the labeling issues in goal and plan recognition, the comparison of goal vs. plan recognition in terms of the pattern length, and the generalizability of the proposed methods. The encouraging results suggest that automated labeling is an important area for future work to further improve language model-based player goal recognition.

A Hyperparameter Search for the Baselines

In the baselines, for LSTM models, we searched the batch size [64, 128] and epoch [50, 100] with a single LSTM hidden layer having 100 hidden units and 0.5 dropout. For GNB, the prior probabilities of the classes were automatically adjusted according to the data. The MLP parameters are {activation: (identity, logistic, tanh, relu), hidden_layer_sizes: [(5, 2), (15,), (100,)]}. The SVM parameters are {estimator__kernel: (linear, rbf), estimator_C: [0.1, 1, 10]}. The RF parameters are {criterion: (gini, entropy), max_features: (sqrt, log2), n_estimators: [10, 30, 50, 100], max_depth: [4, 5, 6, 7, 8, 10]}. The LR parameters are {estimator__penalty: (l2, none), estimator__solver: (newton-cg, lbfgs, sag, saga), estimator__class_weight: (balanced, None)}.

Acknowledgments

This work is supported by the National Science Foundation under award DRL-2112635. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Amado, L. R.; Mirsky, R.; and Meneguzzi, F. 2022. Goal Recognition as Reinforcement Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Anil, C.; Wu, Y.; Andreassen, A.; Lewkowycz, A.; Misra, V.; Ramasesh, V.; Slone, A.; Gur-Ari, G.; Dyer, E.; and Neyshabur, B. 2022. Exploring Length Generalization in Large Language Models. arXiv:2207.04901.

Aung, M.; Demediuk, S.; Sun, Y.; Tu, Y.; Ang, Y.; Nekkanti, S.; Raghav, S.; Klabjan, D.; Sifa, R.; and Drachen, A. 2019. The trails of just cause 2: spatio-temporal player profiling in open-world games. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, 1–11.

Baker, C. L.; Tenenbaum, J. B.; and Saxe, R. R. 2007. Goal inference as inverse planning. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 29(1).

Brown, T. B.; Mann, B.; Ryder, N.; et al. 2020. Language Models are Few-Shot Learners. *CoRR*, abs/2005.14165.

Charniak, E.; and Goldman, R. P. 1993. A Bayesian model of plan recognition. *Artif. Intelligence*, 64(1): 53–79.

Chen, M.; Tworek, J.; Jun, H.; et al. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

Chung, H. W.; Hou, L.; Longpre, S.; et al. 2022. Scaling Instruction-Finetuned Language Models. arXiv:2210.11416.

de A. Santos, L. R.; Meneguzzi, F.; Miotto, G. L.; Pereira, R.; and Pereira, A. 2021. An LP-Based Approach for Goal Recognition as Planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Duhamel, T.; Maynard, M.; and Kabanza, F. 2020. A Transfer Learning Method for Goal Recognition Exploiting Cross-Domain Spatial Features. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Fu, Y.; Peng, H.; Sabharwal, A.; Clark, P.; and Khot, T. 2023. Complexity-Based Prompting for Multi-Step Reasoning. arXiv:2210.00720.

Gold, K. 2010. Training goal recognition online from low-level inputs in an action-adventure game. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, 21–26.

Goslen, A.; Carpenter, D.; Rowe, J.; Azevedo, R.; and Lester, J. 2022a. Robust Player Plan Recognition in Digital Games with Multi-Task Multi-Label Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)* , 105–112.

Goslen, A.; Carpenter, D.; Rowe, J.; Henderson, N.; Azevedo, R.; and Lester, J. 2022b. Leveraging Student Goal Setting for Real-Time Plan Recognition in Game-Based Learning. In *Proceedings of the Twenty-Third International Conference on Artificial Intelligence in Education (AIED-22)*, 78–89.

Ha, E.; Rowe, J. P.; Mott, B. W.; and Lester, J. C. 2011. Goal recognition with markov logic networks for playeradaptive games. In *Artificial Intelligence and Interactive Digital Entertainment Conference*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

Huang, J.; and Chang, K. C.-C. 2022. Towards Reasoning in Large Language Models: A Survey. arXiv:2212.10403.

Kautz, H. A.; and Allen., J. F. 1986. Generalized plan recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 32–37.

Kirginas, S.; and Gouscos, D. 2017. Exploring the impact of freeform gameplay on players' experience: an experiment with maze games at varying levels of freedom of movement. *Journal of Serious Games*, 4(4).

Levy, O.; and Goldberg, Y. 2014. Dependency-based word embeddings. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 2: 302–308.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. Online: Association for Computational Linguistics.

Lewkowycz, A.; Andreassen, A.; Dohan, D.; et al. 2022. Solving Quantitative Reasoning Problems with Language Models. arXiv:2206.14858.

Masters, P.; and Vered, M. 2021. What's the context? Implicit and explicit assumptions in model-based goal recognition. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI-21)*. IJCAI Organization.

Mialon, G.; Dessì, R.; Lomeli, M.; et al. 2023. Augmented Language Models: a Survey. *ArXiv*, abs/2302.07842.

Min, W.; Ha, E. Y.; Rowe, J. P.; Mott, B. W.; and Lester, J. C. 2014. Deep learning- based goal recognition in open-ended digital games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, 37–43. Raleigh.

Min, W.; Mott, B.; Rowe, J.; Liu, B.; and Lester, J. 2016. Player Goal Recognition in Open-World Digital Games with Long Short-Term Memory Networks. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, 2590–2596. New York.

Min, W.; Mott, B.; Rowe, J.; Taylor, R.; Wiebe, E.; Boyer, K.; and Lester, J. 2017. Multimodal Goal Recognition in Open-World Digital Games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, 80–86.

Mirsky, R.; Stern, R.; Gal, Y. K.; and Kalech, M. 2016. Sequential Plan Recognition. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI-16)*, 2009–2014. IJCAI Organization.

Mott, B.; Lee, S.; and Lester, J. 2006. Probabilistic goal recognition in interactive narrative environments. In *Proceedings of the National Conference on Artificial Intelligence*, 187–192.

Nguyen, T.-H. D.; Hsu, D.; Lee, W.-S.; and Leong, T.-Y. 2011. CAPIR: Collaborative Action Planning with Intention Recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 61–66.

Nye, M.; Andreassen, A. J.; Gur-Ari, G.; Michalewski, H.; Austin, J.; Bieber, D.; Dohan, D.; Lewkowycz, A.; Bosma, M.; Luan, D.; Sutton, C.; and Odena, A. 2021. Show Your Work: Scratchpads for Intermediate Computation with Language Models. arXiv:2112.00114.

Pereira, R. F.; Pereira, A. G.; and Meneguzzi, F. 2019. Landmark-enhanced heuristics for goal recognition in incomplete domain models. In *Proceedings of the International Conference on Automated Planning and Scheduling*, 329–337.

Rabkina, I.; Kantharaju, P.; Wilson, J. R.; Roberts, M.; and Hiatt, L. M. 2022. Evaluation of Goal Recognition Systems on Unreliable Data and Uninspectable Agents. *Frontiers in Artificial Intelligence*, 4.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *JMLR*, 4.

Rajani, N. F.; McCann, B.; Xiong, C.; and Socher, R. 2019. Explain Yourself! Leveraging Language Models for Commonsense Reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4932–4942. Florence, Italy: Association for Computational Linguistics.

Ramırez, M.; and Geffner, H. 2011. Goal recognition over POMDPs: Inferring the intention of a POMDP agent. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-11)*, 2009–2014. IJCAI Organization.

Schmidt, C.; Sridharan, N.; ; and Goodson, J. 1978. The Plan Recognition Problem: An Intersection of Psychology and Artificial Intelligence. *Artif. Intelligence*, 11(1): 45–83.

Shvo, M.; and McIlraith, S. A. 2020. Active Goal Recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9957–9966.

Srinivasa-Desikan, B. 2018. *Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python*. Gensim, spaCy, and Keras. Packt Publishing Ltd.

Sukthankar, G.; Goldman, R.; Geib, C.; Pynadath, D.; ; and Bui, H. 2014. *An Introduction to Plan, Activity, and Intent Recognition*. Elsevier.

Suzgun, M.; Scales, N.; Schärli, N.; Gehrmann, S.; Tay, Y.; Chung, H. W.; Chowdhery, A.; Le, Q. V.; Chi, E. H.; Zhou, D.; and Wei, J. 2022. Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them. arXiv:2210.09261.

Thoppilan, R.; Freitas, D. D.; Hall, J.; et al. 2022. LaMDA: Language Models for Dialog Applications. arXiv:2201.08239.

Treger, S.; and Kaminka, A. 2022. Towards Computational Modeling of Human Goal Recognition. *Front. Artif. Intell.*, 4.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. In *Proceedings of the 31th Conference on Neural Information Processing SystemsNeurIPS*.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. arXiv:2203.11171.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; brian ichter; Xia, F.; Chi, E.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*.

Yannakakis, G. N.; Spronck, P.; Loiacono, D.; and André, E. 2013. Player Modeling. In Lucas, S. M.; Mateas, M.; Preuss, M.; Spronck, P.; and Togelius, J., eds., *Artificial and Computational Intelligence in Games*, volume 6 of *Dagstuhl Follow-Ups*, 45–59. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-939897-62-0.

Zhao, W. X.; Zhou, K.; Li, J.; et al. 2023. A Survey of Large Language Models. arXiv:2303.18223.

Zhou, D.; Schärli, N.; Hou, L.; Wei, J.; Scales, N.; Wang, X.; Schuurmans, D.; Cui, C.; Bousquet, O.; Le, Q.; and Chi, E. 2023. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. arXiv:2205.10625.

Zhu, J.; and Ontanon, S. 2020. Player-Centered AI for Automatic Game Personalization: Open Problems. In *Foundations of Digial Games (FDG-20)*.