AAAI Spring Symposium Series (SSS-25)

Wu S, Oltramari A, Francis J, Giles CL, Ritter F (2025). LLM-ACTR: from cognitive models to LLMs in manufacturing solutions. In: Proceedings of the 39th Annual Association for the Advancement of Artificial Intelligence (AAAI) Conference on Artificial Intelligence Spring Symposium on Machine Learning and Knowledge Engineering for Trustworthy Multimodal and Generative AI. 340-349.

LLM-ACTR: from Cognitive Models to LLMs in Manufacturing Solutions

Siyu Wu[1]*, Alessandro Oltramari[2,3], Jonathan Francis[2,3], C. Lee Giles[1], Frank E. Ritter[1]

[1]College of Information Sciences and Technology, The Pennsylvania State University; University Park, PA USA
[2]Bosch Center for Artificial Intelligence; Pittsburgh, PA, USA
[3]Carnegie Mellon University; Pittsburgh, PA, USA
{sfw5621, clg20, frank.ritter}@psu.edu, {alessandro.oltramari, jon.francis}@us.bosch.com

Abstract

Using off-the-shelf large language models (LLMs) in manufacturing decision-making often results in broadly competent but noisy behavior. Previous approaches that employ LLMs for decision-making struggle with complex reasoning tasks that require deliberate cognition over fast and intuitive inference. These approaches often report issues related to insufficient grounding, such as human-level but unhuman-like behaviors. Here, we move toward addressing this gap and ask whether language models can learn from cognitive models for human-like decisions. We introduce VSM-ACTR 2.0, an ACT-R cognitive model for manufacturing solutions, and `LLM-ACTR`, a developing framework for knowledge transfer from cognitive models to language models. The ACT-R cognitive architecture is designed to computationally model the internal mechanisms of human cognitive decision-making. `LLM-ACTR` extracts knowledge from ACT-R's internal decision-making processes, represents it as latent neural representations, and injects this content vector into trainable LLM adapter layers. It then fine-tunes the LLMs for downstream decision-making predictions. We find that, after fine-tuning and adding the content vector to the activations during the LLM forward pass, the LLM offers better representations of human decision-making behaviors on a novel Design for Manufacturing problem, compared to an LLM-only model that employs chain-of-thought reasoning strategies. Taken together, the results open up new research directions for equipping LLMs with the necessary knowledge to computationally model and replicate the internal mechanisms of human cognitive decision-making.

Code — https://github.com/SiyuWu528/llm-actr

Introduction

The goal of Industry 4.0 is to create ”intelligent factories” where technologies enable smart decision-making through cooperation among humans, machines, and sensors, exemplified by smart scheduling using sensor data (Zhong et al. 2017; Serrano-Ruiz, Mula, and Poler 2021). A value stream map (VSM) is vital for smart scheduling in manufacturing (Rahani and Al-Ashraf 2012), but plant managers often struggle with its applications due to its intertwined variables. When they resort to off-the-shelf language models for solutions, it often leads to unhuman-like and noisy predictions (Makatura et al. 2024). These challenges hinder optimal decision-making in production management.

Toward trustworthy decision-making by LLMs in manufacturing, we ask whether language models can learn from cognitive models for human-like decisions. This paper proposes `LLM-ACTR` as an initial solution. `LLM-ACTR` builds upon VSM-ACTR 2.0 (hereafter referred to as VSM-ACTR), which was developed from VSM-ACTR 1.0 (Wu, Oltramari, and Ritter 2024). This is an ACT-R cognitive model that simulates human-like decision-making behavior using domain knowledge from VSM. ACT-R, a representative cognitive architecture (CA) (Laird 2012; Anderson et al. 2004), encompasses perception, memory, goal-setting, and action, and has been pivotal in developing synthetic agents for learning and training, e.g., (Anderson et al. 2019; Martin, Gonzalez, and Lebiere 2004). VSM-ACTR executes tasks that mimics human decision-making behaviors, retrieves similar knowledge representations, and simulates the reinforcement learning processes as decision-makers progress from novice to expert. `LLM-ACTR` learns ACT-R model domain knowledge with LLMs for decision-making tasks. The knowledge transfer methods follow the hypothesis of how LLMs generate the next token prediction: LLMs calculate semantically meaningful primitives in the early layers of the residual stream, which are converted into a high-level execution plan in the middle layers, and then into concrete tokens in the final layers (Vaswani et al. 2017; Brown et al. 2020; Raschka 2024).

`LLM-ACTR` (Fig. 1) uses the content vector representation of ACT-R model's full decision-making reasoning steps (data part a), along with the decisions of the cognitive model on the task at scale (data part b). The content vector $\theta \in R^d$ is added to an early layer ℓ_{target} of the residual stream of the transformer. θ was obtained through semantic extraction and dimension reduction of the embedding space of cognitive decision-making steps, aiming to introduce meaningful early perturbation through differentiate activation of layer ℓ_{target}. Using the modified LLM as the base model, it accesses the last contextualized embedding and obtains the masked embedding. A classification layer is then added on top for fine-tuning with the ACT-R model's human-like decisions(data part b).

`LLM-ACTR` transfers ACT-R model's decision-making

*Work done during an internship at Bosch R&T center.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

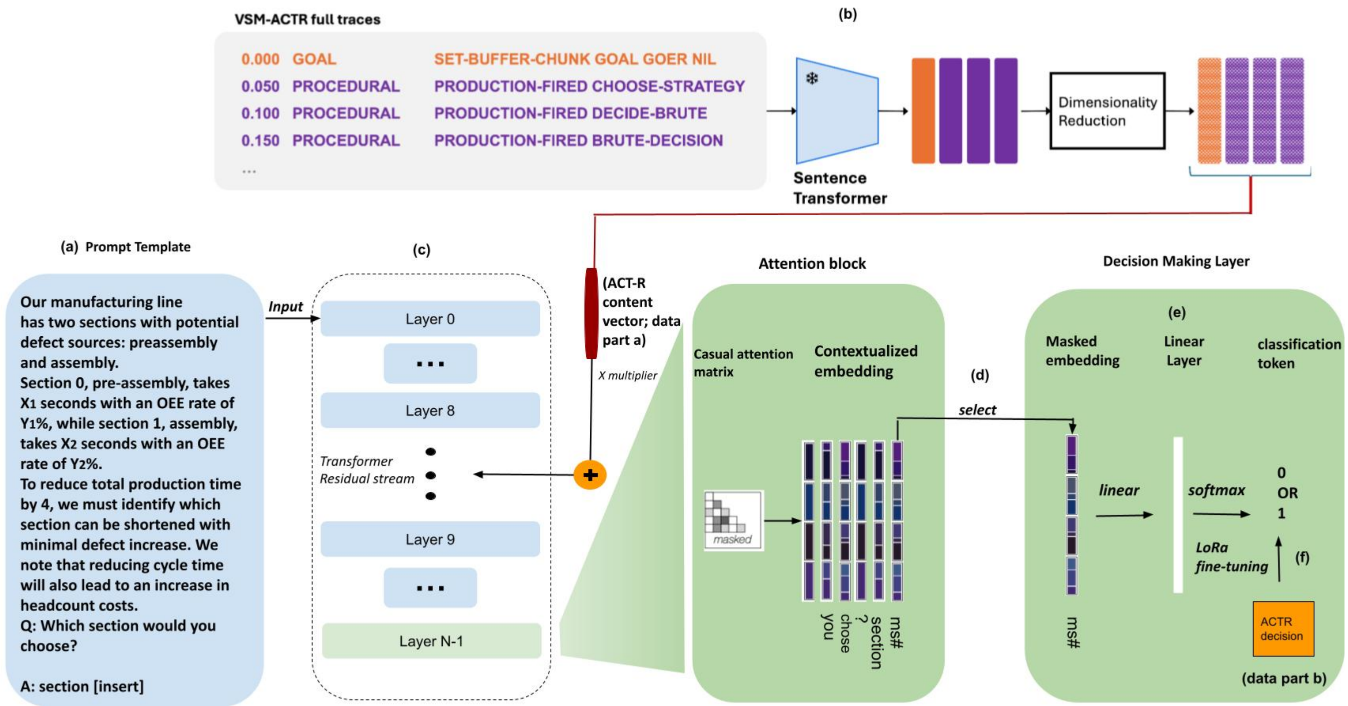

Figure 1: LLM−ACTR begins with (a) parsing consistent template prompts that reflect the decision-making task into an open-source LLM, thereby aligning the task with the ACT-R model. (b) The content vector of the ACT-R model's decision-making traces (data part a) is obtained by passing traces through a sentence transformer to obtain semantic embeddings for each timestamp. Each step's embedding then undergoes dimensionality reduction to retain the primary components before being concatenated into a one-dimensional vector. (c) The cognitive decision-making vector is then injected into the residual stream with a multiplier at one of the layers of the LLM to introduce differentiated activations. (d) Using the modified LLM as the base model, it then accesses the last contextualized embedding and obtains the masked embedding. (e) A classification layer with softmax activation is constructed on top to form the decision-making layer. (f) Using targets of ACT-R model decisions (data part b), the LLM is fine-tuned for the classification task in decision-making using the Low Rank Adaptation method (LoRA).

knowledge to an LLM on the task. It leverages the strengths of both LLMs and CAs by using the natural language processing and generative capabilities of LLMs, complemented by the human-like learning and reasoning offered by CAs. We then present a case study of LLM−ACTR in manufacturing decision-making. The task is associated with a key aspect of Design For Manufacturing (DFM): enhancing product development and optimizing production system performance by improving time efficiency and reducing headcount costs (Ulrich et al. 1993) (Fig.1(a) Prompt Template).

The following sections detail the VSM-ACTR cognitive model and its real-time reasoning traces on the task, which are then converted into domain knowledge for LLM decision augmentation. This is followed by an overview of the LLM−ACTR, the experiments, results, and discussions.

Related Work

Relating Cognitive Psychology to Human-Like Artificial Intelligence

Human-like artificial intelligence (HLAI) has been a goal since the emergence of machines (McCarthy 2007). In recent years, the development of transformer-based LLMs has revolutionized HLAI, demonstrating impressive human-level capabilities. However, LLMs sometimes fail to display human-like behavioral traits. Analyzing the areas where LLMs currently fall short in replicating human cognition and behavior highlights the problems in exhibiting human-level capabilities that are unhuman-like (Dorobantu 2021), including behavior discrepancies between LLM inference and human reasoning (Binz and Schulz 2023; Liu et al. 2024), insufficient grounding (Yao et al. 2023), and hallucination (Chakraborty, Ornik, and Driggs-Campbell 2024).

The challenges mentioned have catalyzed an integration of cognitive psychology with LLMs, toward human-like trustworthy LLMs. Recent studies have used cognitive psychology experiments to investigate and comprehend behaviors in these models, focusing more on behavioral insights than on conventional performance metrics (Binz and Schulz 2023; Coda-Forno et al. 2024b). In addition, the use of LLMs' neural representations has been applied in behavioral psychological science research, which involves but is not limited to prompt engineering, feature extraction, and fine-tuning, which we cover next.

Feature Extraction. The process begins with passing text that mirrors a psychological experiment through the open-source LLM to capture contextualized embeddings from the final layer (Hussain et al. 2024). These embeddings can be employed in various psychological experiments applications, such as predicting similarities between personality constructs (Abdurahman et al. 2023), choices in reinforcement learning (Binz and Schulz 2024), or perceptions related to risk or health (Wulff and Mata 2023). For tasks that require sequence prediction, decoder models are preferred due to their larger size and more extensive training data (Hussain et al. 2024).

Zero-shot and Few-shot Learning. Zero-shot learning enables the creation of categorical or numerical predictions, such as evaluating sentiments on social media (dos Santos et al. 2024), without requiring training specific to the task. Few-shot learning extends this concept by adding minimal supervision, such as a small number of example pairs, to improve the accuracy of the model.

Fine-Tuning. Fine-tuning smaller LLMs for human-like behaviors can achieve performance that matches or exceeds that of larger models under zero- or few-shot learning conditions (Hussain et al. 2024). This involves adjusting model weights to improve task-specific outcomes. For example, one study fine tuned BERT in zero-shot learning to predict reinforcement learning behaviors of human subjects (Hussain et al. 2024). However, the generalization of this approach is challenged by the high cost of collecting large cognitive psychological datasets involving human subjects.

Common Model of Cognition, Cognitive Architectures, and Cognitive Model

Toward integrating human-like behavioral traits into LLMs, we use a suite of tools rooted in the Common Model of Cognition (CMC) to provide a wider range of tasks into the training dataset. CMC implements a unified Theory of Mind (Newell 1994; Laird, Lebiere, and Rosenbloom 2017), a theoretical framework that presents a model of human cognition codified as a computational architecture. The CMC is a brain-inspired framework validated by large-scale neuroscience data. The CMC identifies core components and processes fundamental to human cognition, including memory, perception, motor actions, and decision-making. The model assumes a cyclical process where these components interact to produce human behavior. The CMC includes a feature-based declarative long-term memory, a buffer-based working memory, a system for pattern-directed action invocation stored in procedural memory, and specialized systems for perception and action (Stocco et al. 2021).

The CMC integrates essential features from various cognitive architectures (CA), which are computational frameworks designed to capture the invariant mechanisms of human cognition. These mechanisms include functions related to attention, control, learning, memory, adaptivity, perception, and action (Laird 2012; Anderson et al. 2019). ACT-R is one of the representative CA designed to model human behaviors (Anderson 2009). ACT-R models can store, retrieve, and process knowledge, as well as explain and predict performance (Ritter, Tehranchi, and Oury 2019; Bothell 2017).

There are currently two kinds of knowledge representations in ACT-R, and they are declarative knowledge and procedural knowledge. Declarative knowledge consists of chunks of memory (e.g., the production line comprises five sections), while procedural knowledge performs basic operations, moves data among buffers, and identifies the next instructions to be executed (e.g., lower defect rate will lead to higher efficiency rate). ACT-R has been widely applied to build models that automate decision-making tasks across psychology and computer science, e.g., (Blessing and Anderson 1996; Wu et al. 2023).

However, ACT-R models do not accept natural language as input and cannot easily generalize across problems, which limits their flexibility for decision-making. Intuitively, a solution could take the best of both CAs and LLMs, e.g., (Sumers et al. 2024; Zhu and Simmons 2024), where ACT-R models serve as synthetic agents to instruct LLMs. They do this by providing knowledge of cognitive decision-making through LLMs' training, which includes aspects such as learning. The trained LLMs can then be generalized to unseen problems.

Problem Definition: Design for Manufacturing

This paper presents a case study of training a cognitively inspired LLM for decision-making in the design for manufacturing (DFM) domain. We define the terminology that constitutes our decision-making problem. The DFM problem setting is a prototypical manufacturing production-line workflow, from supplier to customer, for which there exists a VSM (Fig. 2), which allows for tracking the efficiency at different sectors of the process and abstracts the overall problem for mathematical modeling and optimization. Decision candidates come from sectors such as Body Production, Pre-Assembly, Assembly. Early sectors pose potential efficiency problems in the workflow and may warrant optimization (triangles), while later stages are governed by First-In-First-Out (FIFO) processes. The metrics at each stage include Cycle Time (CT), Overall Equipment Effectiveness (OEE), and/or Mean Absolute Error (MAE). Focused on maintaining stable output for manufacturing plants, we consider plant managers' feedback alongside the VSM structure to define the decision-making problem that aim to reduce total production time while minimizing total defect rate increase(see Fig.1(a) for Prompt Template). When facing unseen DFM problems, which are yet constrained to fixed decision candidates and unknown decision metrics. `LLM-ACTR` takes a natural language question prompt, and outputs a binary decision (0 or 1) on which of two sectors, pre-assembly or assembly, requires a time reduction.

VSM-ACTR, A Human-Like Decision Making Cognitive Model with Metacognition for Manufacturing Solutions

To investigate whether LLMs can learn from cognitive models for human-like decisions, we start by creating the cognitive model for DFM problem.

Human Centered Model Design

We built VSM-ACTR, which is a rule-based ACT-R cognitive decision-making model for DFM problem that imple-

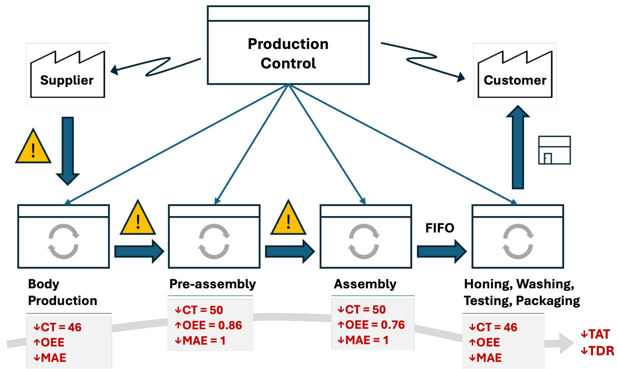

Figure 2: Value Stream Map, the yellow triangles show the possible optimization sectors.

ments multiple problem-solving strategies, through a combination of production rules.

VSM-ACTR has incorporated the meta-cognitive processes that reflect on and evaluate the progress of chosen strategies—with an emphasis on headcount cost evaluation, through a reward structure that enables a process akin to reinforcement learning. This system enables the model to dynamically assess the impact of decisions on headcount costs, computing a reward or penalty for each decision cycle. These rewards or penalties then dynamically adjust the utility of the productions associated with each decision-making cycle. This helps the model to exhibit a human-like learning progression.

VSM-ACTR integrates the prototypical decision process with insights into how cognitive models represent different levels of expertise, e.g., (Martin, Gonzalez, and Lebiere 2004; Blessing and Anderson 1996), categorizing users into three levels of expertise: novices, intermediates, and experts. Novices engage in decision-making using intuitive deliberative chunks. Intermediates can manage key metrics such as CT and OEE but struggle with the systematic analysis of intertwined variables. Experts, on the other hand, make judgments systematically. The cognitive model employs three types of knowledge chunks: decisions, decision merits, and goals. The 'decision chunk' encodes eight slots including reduction time (goal), decision-making state (novice, intermediate, expert), and related variables. The 'decision merits chunk' holds information on sector weights, defect increases by sector, and comparative defect rate increases. The 'goal chunk' captures the initial production conditions and the ultimate goal of achieving the optimal decision. In addition, the model uses 18 procedural rules driven by goal-focused objectives across 20 states, covering actions such as choosing strategies, actions, working memory management, decisions, and evaluations.

Level of Expertise Mechanism

The model can learn while performing tasks through a mechanism leading to varying levels of expertise, as shown in Figure 3. The model mimics human decision-making behavior through differentiating knowledge representations. **Declarative Memories**: These memories store knowledge that aligns with human intuition and expertise gained from the VSM. For example, the green triangles in the figure represent a portion of the intuition used by novice decision-makers, while the red circles contains VSM domain knowledge used by intermediate decision-makers. **Production Rules**: These rules capture the rational decision-making processes observed in human subjects. The green lines illustrate how the imaginal buffer retrieves relevant portions of the novice declarative memory and feeds them to the novice production rule set. Intermediate and expert decision-making levels follow the same principle. Red and blue shapes represent their respective declarative memory chunks, and the corresponding colored arrows show the flow of information through their production rule sets. Finally, the goal buffer uses the 'goal focus' command to manipulate the different phases of the task.

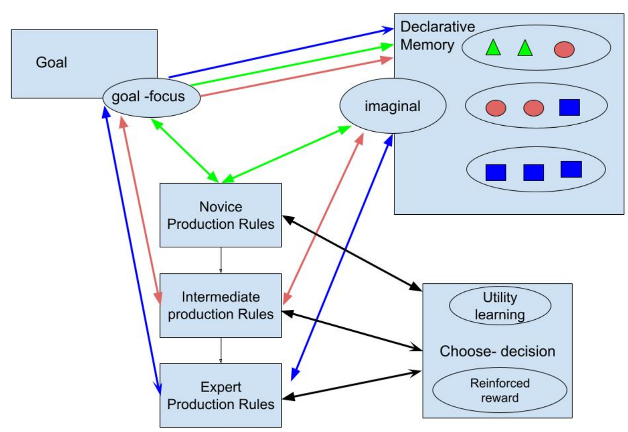

Figure 3: Level of expertise mechanism in VSM-ACTR.

The model also simulates the learning progress through the **Decision-Choice Control**, which manages errors, learning, and memory via utility learning and reinforced rewards. Novice decision-making productions start with a utility base and include a noise setting. Each round of decisions receives rewards or penalties, and the utility of associated production rules updates with the adjustment of memory retention, which depends on the time passed since the rule last fired.

Foster Metacognition to Support Learning

With the aim of making the model assess the effectiveness of decisions while learning — akin to human metacognition, self-assessing and self-correcting in response to self-assessment (Nelson and Narens 1994) — we consequently developed a dynamic reward function that rewards actions after self-evaluating the chosen strategy.

VSM-ACTR uses a Temporal Difference (TD) algorithm from reinforcement learning (Sutton and Barto 1999) as expressed in Eqn. 1. Each production rule in the ACT-R model has a utility—a value or strength—associated with it, which is updated using the TD algorithm:

$$Eqn.\,1:\; U_i(n) = U_i(n-1) + \alpha\left[R_i(n) - U_i(n-1)\right]$$

where $U_i(n)$ represents the value or utility of some item i (e.g., a production) after its n-th occurrence, and $R_i(n)$ represents the reward received on the n-th occurrence. The parameter α ($0 < \alpha < 1$) controls the learning rate. If multiple productions compete with expected utility values U_j, the probability of of selecting production i is given by Eqn. 2:

$$Eqn.\,2:\ \text{Probability}(i) = \frac{e^{U_i/\sqrt{2s}}}{\sum_j e^{U_j/\sqrt{2s}}},$$

where the summation over j is over all the productions that currently have their conditions satisfied; and s is the noise.

The utilities of production are learned as the model runs, based on the rewards or penalty that are received. Where we designed the reward function as $R(s, f(x))$ that calculates the reward at the end of each decision-making round. This function takes two parameters: S, representing the strategy used, and $f(x)$, which results from headcount cost analysis, leading to either a weighted reward or a penalty. For example, in one decision round, a penalty of -2 is computed due to the use of a novice strategy coupled with inefficient headcount cost analysis. Factoring in the memory retention effect after a 0.05 seconds step, the calculation using the TD algorithm modifies the impact of the decision on the utility of the next production as:

$$U_i(n+1) = U_i(n) + \alpha\,[-2 - 0.05 - U_i(n)]\,.$$

This will then sequentially update the utility of the chain of productions for the chosen strategy. We find that when the model encounters certain types of problems where both novice and expert strategies result in similar efficiencies in cost assessment. In these cases, the model is prone to staying with the novice strategy and exhibits a more gradual learning curve, similar to the tendency for people facing bounded rationality in decision-making (Hastie and Dawes 2010), where they are likely to select the less effortful option when faced with multiple choices that produce very similar outcomes.

Data Collection and Evaluation

We then explain the decision-making knowledge curated from the real-time reasoning steps of VSM-ACTR, followed by data collection and evaluation.

VSM-ACTR Knowledge Representations

This study curated VSM-ACTR domain knowledge through VSM-ACTR's traces, which capture the reasoning steps in real time using a concurrent protocol. These traces log the cognitive operations executed by various modules at each decision point. The traces exhibit metacognition, which involves awareness and understanding of one's own decision-making processes. This is represented through model traces that demonstrate the use of the imaginal buffer for accessing working memory, procedural memory matching and firing, and the self assessment of strategy effectiveness. Traces also exhibit executive function (Gilbert and Burgess 2008), which involves the evolution of decision-making results across trials and shows how decisions adapt through learning and experience.

As shown in Table 1, the model begins by establishing the goal (line 1) and then proceeds with a novice strategy (line 3, BRUTE). For the production rules associated with each strategy, the utility of each production rule is updated based on the received reward and the time since the last selection. For instance, the reward computation based on cost analysis (line 6) for the BRUTE choice results in a reward of -2 (line 10). Consequently, the utility of the NAIVE-CHOICE rule, impacted by a penalty of -2.25 for the time passed since the last selection, decreases from 3 to 1.96 (lines 14-16). As the utility of naive strategies declines, the probability of triggering the Intermediate Strategy (lines 26-27) and the EXPERT Strategy (lines 87-89) increases.

001	0.000 GOAL SET–BUFFER–CHUNK GOAL GOER NIL
002	0.050 PROCEDURAL PRODUCTION–FIRED CHOOSE–STRATEGY
003	0.100 PROCEDURAL PRODUCTION–FIRED DECIDE–BRUTE
004	0.150 PROCEDURAL PRODUCTION–FIRED BRUTE–DECISION
005	assembly is always a good place to reduce time!
006	0.200 PROCEDURAL PRODUCTION–FIRED HEAD-COUNT
007	-0.01999998
008	0.250 PROCEDURAL PRODUCTION–FIRED STOP
009	this is the end of one decision making
010	Utility updates with Reward = -2.0 alpha = 0.2
011	Updating utility of production CHOOSE–STRATEGY
012	U(n-1) = 0.0 R(n) = -2.25 [-2.0 - 0.25 seconds since selection]
013	U(n) = -0.45000002
014	Updating utility of production DECIDE–BRUTE
015	U(n-1) = 3.0 R(n) = -2.2 [-2.0 - 0.2 seconds since selection]
016	U(n) = 1.96
...	...
026	0.300 PROCEDURAL PRODUCTION–FIRED CHOOSE–STRATEGY
027	0.350 PROCEDURAL PRODUCTION–FIRED DECIDE–INTERMEDIATE
...	...
056	0.800 PROCEDURAL PRODUCTION–FIRED CHOOSE–STRATEGY
057	0.850 PROCEDURAL PRODUCTION–FIRED EXPERT–STRATEGY
...	...
084	Updating utility of production CHOOSE–STRATEGY
085	U(n-1) = -0.46 R(n) = 4.65 [6.0 - 1.35 seconds since selection]
086	U(n) = 0.56200004
087	Updating utility of production EXPERT–STRATEGY
088	U(n-1) = 0.0 R(n) = 4.7 [6.0 - 1.3 seconds since selection]

Table 1: VSM-ACTR decision-making traces that highlight goal initiation, strategy selection, decision evaluation, utility update, and learning.

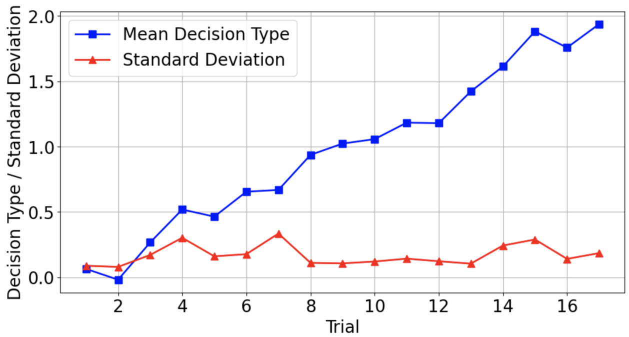

Figure 4: Trend of decision types over trials, blue line is average decision types, red line is variance.

Data Collection

With the VSM-ACTR persona's decision-making traces in hand, the next step involves converting these traces into data to train LLMs. As shown in Fig. 1, the data contains two parts: Part A is a vector that learns the embedding space of the model's decision-making steps; Part B consists of model decisions across trials.

To create data part a, this study employs a semantic extraction and dimension reduction approach. This approach aims to transform a vast number of cognitive reasoning stamps into a vector format that balances information retention with computational efficiency. Traces for each task are processed through a sentence transformer to obtain semantic embeddings for each timestamp. A Sum of Ranked Explanatory Effects (SREE) analysis is then applied to determine the number (N) of principal components that account for at least 70% of the variance. These embeddings are then reduced to N dimensions using Principal Component Analysis (PCA) (Abdi and Williams 2010) (Fig. 1(b)). The reduced embeddings for each timestamp are then concatenated into a one-dimensional vector. To create data part b, this study logs decisions, which are then numerically encoded. '0' represents a decision to reduce time in the preassembly section, and '1' indicates a decision for assembly. These data are subsequently serving as the targets for fine-tuning (Fig. 1(f)).

Data Evaluation

Processed data is used as domain knowledge to train LLMs; therefore, the next step is to evaluate the quality of the data.

Use Semantic Mapping to Evaluate Cognitive Decision Making Traces Vector To answer the question of whether the vector learns an embedding space of decision traces (data part a), this study conducted a semantic mapping analysis of the first two principal components of the learned embeddings. Figure 5 shows the reduced embeddings corresponding to the semantic mapping of ACT-R's components, including procedural memory, imaginal memory, goal knowledge, utility updating, and decision-making actions.

The MANOVA analysis was conducted to assess the overall effect of the independent variables, which include label categories or ACT-R components, on the combined dependent variables—components of reduced embeddings. This analysis reveals a significant relationship with the semantic mapping of ACT-R's components. For instance, the Wilks' lambda value (0.0004) suggests that the label or ACT-R component categories explain nearly all the variance in the dependent variables, indicative of a strong group effect. The statistical tests applied—Wilks' lambda, Pillai's trace, Hotelling-Lawley trace, and Roy's greatest root—all demonstrate strong significance, as evidenced by p-values less than 0.05 across all tests. It shows that the semantics of neural symbolic representations can be learned using sentence transformer, and the principal components retained successfully capture the essential variance related to these cognitive processes.

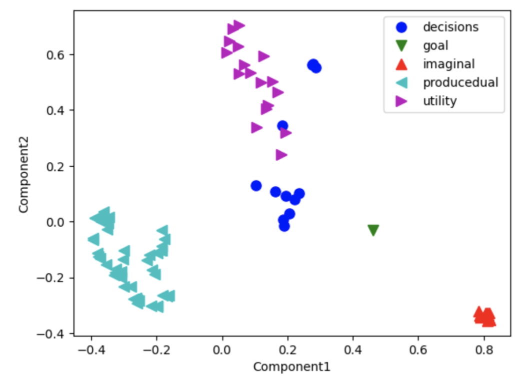

Figure 5: Reduced embedding map to full traces from VSM-ACTR one trial.

Analyze Repeated Reinforcement Decisions To answer the question of whether VSM-ACTR decisions (data part b) demonstrate learning progression, and capture individual differences, this study first uses descriptive statistics and linear regression to show the average progression of decision types across trials. It then use a mixed linear model to assess and illustrate the effects of trials on decision types across ACT-R model personas, with repeated measures of trials, and random effects to account for individual differences. Last but not least, use ordered logistic regression to analyze and understand the relationship between the number of trials and an ordinal dependent variable of learning progress from novice to expert.

We ran the VSM-ACTR model 2,012 times to understood its behavior (Ritter et al. 2011). Each time, we asked it to run 15-16 trials until the model achieved stable expert behavior. We collected data with decision types encoded as 0, 1, and 2 for novice, intermediate, and expert strategies.

Fig. 4 shows a significant positive impact of trial exposure on decision-making progression, evidenced by a coefficient of 0.086 ($p < 0.05$). furthermore, the standard deviation starts relatively low but quickly increases, peaking around the third trial. This could reflect a diverging approach to decision-making as VSM-ACTR personas experiment with different strategies. the standard deviation gradually decreases thereafter, stabilizing between 0.5 and 0.75, which points to a convergence in decision-making strategies among personas. A mixed linear model regression confirms the effect of trials on decision-making and further reveals a variance of 0.007 in the random group effects, suggesting that the trials themselves predominantly explain the variability in decision type, while the individual differences exists. Threshold analysis using ordered logistic regression reveals significant transition thresholds. The transition from novice to intermediate has a significant threshold of 0.88 ($p < 0.05$), indicating a challenging progression to higher decision-making skills. In contrast, the transition from intermediate to expert shows a significantly lower threshold of 0.1 (p = 0.021), suggesting it is easier to progress from intermediate to expert than from novice to intermediate. These findings validate that the repeated reinforcement decisions from VSM-ACTR demonstrate human like learning progression and capture individual differences.

LLM-ACTR Framework, Experiments, and Results

With the validated data from VSM-ACTR in hand, this section begins by explaining the design principles of the `LLM-ACTR` framework, followed by the experiments, results, and discussions.

Design Principles of the LLM-ACTR Framework

Figure 1 shows `LLM-ACTR`. With the goal of knowledge transfer in mind, we designed the `LLM-ACTR` based on the mechanism of LLMs' next-token prediction and the knowledge representation of VSM-ACTR.

During the early layers, when LLMs calculate semantically meaningful primitives. We drew inspiration from activation engineering for LLM behavior control, e.g., (Turner et al. 2023; Zou et al. 2023), where the learned content vector is added to the residual stream of the layer to steer behavior toward desired outcomes. Instead of learning the content vector from contrast pairs of training data, e.g., (Panickssery et al. 2023; Xu and Wang 2023), which is not applicable in our case, we develop a cognitive decision-making concept vector through semantic extraction and dimensionality reduction using the VSM-ACTR decision-making traces (see Table 1). We then added it to the activations during the forward pass to elicit meaningful behavior perpetuation (Subramani, Suresh, and Peters 2022). This was then converted into a high-level execution plan in the middle layers. During the final stages of the model head, where concrete decision tokens are generated, we drew inspiration from fine-tuning for transfer learning, e.g., (Swati et al. 2019; Kim et al. 2022), and the recent findings that LLMs' final layer of contextualized embeddings contains rich neural representations for predicting human behaviors (Binz and Schulz 2024; Wulff and Mata 2023). Therefore, we access the last contextualized embedding, obtain the masked embedding, and fine-tune the LLM with the target as human-like decisions.

Base Model and Data

The case study uses LlaMa-2 7B model as the base model because it demonstrated effectiveness and efficiency in NLP tasks (Huang, Hu, and Wang 2024). As a state-of-the-art LLM, LlaMa has been trained on trillions of tokens from publicly available datasets. Unlike other transformer-based models such as the GPT family, which can only be accessed at the user's end, LlaMa's architecture, including its pre-trained weights, is fully accessible. Furthermore, evidence that its internal representations can be trained to become more aligned with human neural activity has been presented (Binz and Schulz 2024).

The VSM-ACTR full trace vector (data part A) is obtained by processing random 240 ACT-R persona traces using semantic extraction and dimension reduction. Note that each persona has vectors of varied lengths due to the simulated individual differences. We addressed the issue of ragged tensors by padding, then calculated the standardized mean values of these vectors, and integrated the normalized vector into the residual stream of one of the early hidden layers of the transformer, using a scaling factor (multiplier) to control the magnitude of the vector's effect.

To determine the data part b size that can effectively perform the fine-tuning task while balancing efficacy and resource limitations, we referred to (Kumar, Sharma, and Bedi 2024), who showed evidence that LlaMa-2 7B would maintain competitive performance in resource-limited text classification with datasets of nearly 1,000 rows per class. Based on this, we created a dataset that contains 2,012 decision-making trials, obtained by running the developed VSM-ACTR model across 32 problem sets, each ACT-R persona was run for 15-16 trials until more stable expert behavior was achieved.

Experiment Evaluation

To assess the model's ability to make human-like decisions, we first split the data into train and validation sets to reserve a set of unseen problems. We then compared the predictive negative log-likelihood (NLL), a measure of goodness-of-fit, of `LLM-ACTR` in predicting VSM-ACTR's decisions on the unseen problems, against a pre-trained LlaMa and a LlaMa fine-tuned without content vectors. Using LlaMa without fine-tuning provides a measure of the overall impact of `LLM-ACTR` on learning human-like decisions, and using a fine-tuned LlaMa offers insights into the magnitude of the impact on task performance from the cognitive decision-making vector compared to fine-tuning with reinforced decisions.

Experiment Metrics

The VSM-ACTR full trace vector is set to be trainable when added to the residual stream in ℓ_{target}. The fine-tuning process of `LLM-ACTR` and the fine-tuning of LlaMa-only use the same metrics. The fine-tuning employs cross-entropy as the loss function and uses Adam optimization. Training involves a train-test split of 0.2 and uses a batch size of 5 for both training and validation phases. The learning rate is set to 1×10^{-5}, with training spanning across 10 epochs. To ensure regularization and prevent overfitting, a weight decay of 0.01 and a dropout rate of 0.5 are applied, and gradient accumulation is set to 2. Lastly, gradient clipping is employed to maintain a maximum gradient norm of 1.0 for gradient explosion control.

Experiment Results

The training loss of `LLM-ACTR` begins at 0.85, with fluctuations observed in subsequent epochs and a notable dip at epoch 7 and ended with 0.847 at epoch 10. The validation loss starts at around 0.68 and remains generally reduced throughout the epochs with end up 0.65 at 10 epochs, showing `LLM-ACTR` learns effectively. In addition, the `LLM-ACTR` has an average NLL of 0.65 on held-out data across 10 epochs, compared to the pretrained LlaMa, which has an NLL of 0.904 on held-out data. This shows that LlaMa learns from VSM-ACTR and obtains better predictions of human-like decisions.

We then compared `LLM-ACTR` with fine-tuned-only LlaMa to assess the impact of injecting VSM-ACTR content vecotr into LlaMa's hidden layer on knowledge transfer. The results show that `LLM-ACTR` had an improved NLL compared to LlaMa with fine-tuning only, as illustrated in Fig. 6. Adding the vector representation of VSM-ACTR's full traces during fine-tuning slightly decreased the mean and reduced the variance of NLL across 10 epochs, indicating better model fit and stability compared to fine-tuning alone. The improved model fitting of `LLM-ACTR` suggests that the learned vector from VSM-ACTR's full traces provides additional useful knowledge to LlaMa, enabling it to better capture underlying patterns in human-like decisions.

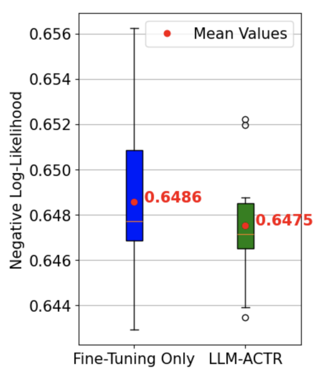

Figure 6: The NLL comparison across 10 epochs of LLM-ACTR against fine-tuning only LLaMa.

Results and Discussion

The results show that after knowledge transfer from a cognitive model to an LLM, through behavior perturbation and fine-tuning, the LLM offers better representations of human behaviors compared to an LLM-only model that employs chain-of-thought reasoning on the DFM task. Furthermore, allowing the model to integrate and learn from the cognitive vector during training potentially leads to more nuanced and human-like decision-making capabilities, as captured by the cognitive features encoded in the vector. However, the influence of the cognitive content vector is limited and warrants further investigation, partly because the stochastic simulation of the VSM-ACTR produces decision-making vectors of various lengths. This study addresses ragged tensors by padding, but this approach potentially dilutes or changes the semantics of each vector. To improve the impact of the cognitive vector, additional techniques such as vector optimization will be needed.

Regardless, this development opens up new research directions for equipping LLMs with the necessary knowledge to computationally model and replicate the internal mechanisms of human cognitive decision-making (Oltramari et al. 2021). It also complements ongoing work showing that LLMs could possibly be transformed into cognitive models through knowledge transfer, e.g., (Binz and Schulz 2024; Coda-Forno et al. 2024a,b). For example, (Binz et al. 2024) demonstrates that through fine-tuning, LLMs' internal representations become more aligned with human neural activity.

Conclusion

Contribution. The present study offers three contributions: (1) It introduces VSM-ACTR, a human-like cognitive model for manufacturing solutions, which has been improved to model metacognitive processes to reflect on and evaluate the effectiveness of the actions. (2) It advances previous efforts on human-like LLMs alignment using data from large-scale cognitive psychology experiments involving human subjects (Binz and Schulz 2023; Coda-Forno et al. 2024a). It reduces the cost of data collection by using synthetic data from cognitive models. The synthetic data present real-time cognitive reasoning with tasks, including metacognition, which is hard to quantify in human subjects (Fleming and Lau 2014). (3) It presents a developing framework of knowledge transfer from cognitive models to language models, rooted in the mechanism of LLMs' next-token prediction and the knowledge representation of cognitive models: one integrates a cognitive decision-making vector into the early layer of the residual stream to elicit meaningful behavior perpetuation (Panickssery et al. 2023), and the other occurs in the later phase of model heads, using the cognitive models' decisions for fine-tuning (Guo et al. 2019). The case study demonstrates that `LLM-ACTR` achieves better fit to human-like decisions on unseen problems compared to a pre-trained model in the DFM task. Thus, it is possible to transfer decision making knowledge from cognitive models to LLMs by adding a cognitive concept vector to the forward pass activation and fine-tuning.

Limitation and Future Work. `LLM-ACTR` can now generalize to unseen problems within an applicable domain, constrained by fixed decision candidates and unknown decision metric values. In applying `LLM-ACTR` to problems that incorporate an increasing number of decision candidates and associated metrics, it becomes critical to solve out-of-domain problems (Wang et al. 2022). This will require `LLM-ACTR` to progress to transferring guided perception, memory, and goal-setting to LLMs. As (Zhu and Simmons 2024) found, training the LLM with the rules of guided perception in cognitive models can help generalize robotics problem-solving to out-of-distribution tasks.

References

Abdi, H.; and Williams, L. J. 2010. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4): 433–459.

Abdurahman, S.; Vu, H.; Zou, W.; Ungar, L.; and Bhatia, S. 2023. A deep learning approach to personality assessment: Generalizing across items and expanding the reach of survey-based research. *Journal of Personality and Social Psychology*, 126(2): 312. Advance online publication.

Anderson, J. R. 2009. *How can the human mind occur in the physical universe?* Oxford University Press.

Anderson, J. R.; Betts, S.; Bothell, D.; Hope, R.; and Lebiere, C. 2019. Learning rapid and precise skills. *Psychological Review*, 126: 727–760.

Anderson, J. R.; Bothell, D.; Byrne, M. D.; Douglass, S.; Lebiere, C.; and Qin, Y. 2004. An integrated theory of the mind. *Psychological Review*, 111(4): 1036–1060.

Binz, M.; Akata, E.; Bethge, M.; Brändle, F.; Callaway, F.; Coda-Forno, J.; Dayan, P.; Demircan, C.; Eckstein, M. K.; Éltető, N.; Griffiths, T. L.; Haridi, S.; Jagadish, A. K.; Ji-An, L.; Kipnis, A.; Kumar, S.; Ludwig, T.; Mathony, M.; Mattar, M.; Modirshanechi, A.; Nath, S. S.; Peterson, J. C.; Rmus, M.; Russek, E. M.; Saanum, T.; Scharfenberg, N.; Schubert, J. A.; Buschoff, L. M. S.; Singhi, N.; Sui, X.; Thalmann, M.; Theis, F.; Truong, V.; Udandarao, V.; Voudouris, K.; Wilson, R.; Witte, K.; Wu, S.; Wulff, D.; Xiong, H.; and Schulz, E. 2024. Centaur: a foundation model of human cognition. *arXiv:2410.20268*.

Binz, M.; and Schulz, E. 2023. Using cognitive psychology to understand GPT-3. *Proceedings of the National Academy of Sciences*, 120(6): e2218523120.

Binz, M.; and Schulz, E. 2024. Turning large language models into cognitive models. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Blessing, S. B.; and Anderson, J. R. 1996. How people learn to skip steps. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(3): 576.

Bothell, D. 2017. *ACT-R 7 reference manual*. Available at http://act-r.psy.cmu.edu/wordpress/wp-content/themes/ACT-R/actr7/reference-manual.pdf.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 1877–1901.

Chakraborty, N.; Ornik, M.; and Driggs-Campbell, K. 2024. Hallucination detection in foundation models for decision-making: A flexible definition and review of the state of the art. *arXiv*, 2403.16527: 31.

Coda-Forno, J.; Binz, M.; Akata, Z.; Botvinick, M.; Wang, J.; and Schulz, E. 2024a. Meta-in-context learning in large language models. In *Advances in Neural Information Processing Systems*, volume 36, 65189–65201.

Coda-Forno, J.; Binz, M.; Wang, J. X.; and Schulz, E. 2024b. CogBench: A large language model walks into a psychology lab. In *International Conference on Learning Representations (ICLR)*.

Dorobantu, M. 2021. Human-level, but non-humanlike: Artificial intelligence and a multi-level relational interpretation of the imago Dei. *Philosophy, Theology and the Sciences*, 8(1): 81–107.

dos Santos, V. G.; Santos, G. L.; Lynn, T.; and Benatallah, B. 2024. Identifying citizen-related issues from social media using LLM-based data augmentation. In *International Conference on Advanced Information Systems Engineering*, 531–546. Cham: Springer Nature Switzerland.

Fleming, S. M.; and Lau, H. C. 2014. How to measure metacognition. *Frontiers in Human Neuroscience*, 8: 443.

Gilbert, S. J.; and Burgess, P. W. 2008. Executive function. *Current Biology*, 18(3): R110–R114.

Guo, Y.; Shi, H.; Kumar, A.; Grauman, K.; Rosing, T.; and Feris, R. 2019. Spottune: transfer learning through adaptive fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4805–4814.

Hastie, R.; and Dawes, R. M. 2010. *Rational choice in an uncertain world: The psychology of judgment and decision making*. Sage.

Huang, D.; Hu, Z.; and Wang, Z. 2024. Performance analysis of Llama 2 among other LLMs. In *2024 IEEE Conference on Artificial Intelligence (CAI)*, 555–559.

Hussain, Z.; Binz, M.; Mata, R.; and Wulff, D. U. 2024. A tutorial on open-source large language models for behavioral science. *Behavior Research Methods*, 56(8): 8214–8237.

Kim, H. E.; Cosa-Linan, A.; Santhanam, N.; Jannesari, M.; Maros, M. E.; and Ganslandt, T. 2022. Transfer learning for medical image classification: A literature review. *BMC Medical Imaging*, 22(1): 69.

Kumar, A.; Sharma, R.; and Bedi, P. 2024. Towards optimal NLP solutions: Analyzing GPT and LLaMA-2 models across model scale, dataset size, and task diversity. *Engineering, Technology & Applied Science Research*, 14(3): 14219–14224.

Laird, J. E. 2012. *The Soar cognitive architecture*. MIT Press.

Laird, J. E.; Lebiere, C.; and Rosenbloom, P. S. 2017. A standard model of the mind: Toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics. *AI Magazine*, 38(4): 13–26.

Liu, R.; Geng, J.; Peterson, J. C.; Sucholutsky, I.; and Griffiths, T. L. 2024. Large language models assume people are more rational than we really are. *arXiv.2406.17055*.

Makatura, L.; Foshey, M.; Wang, B.; Hähnlein, F.; Ma, P.; Deng, B.; Tjandrasuwita, M.; Spielberg, A.; Owens, C. E.; Chen, P. Y.; Zhao, A.; Zhu, A.; Norton, W. J.; Gu, E.; Jacob, J.; Li, Y.; Schulz, A.; and Matusik, W. 2024. How can large language models help humans in design and manufacturing? Part 2: synthesizing an end-to-end LLM-enabled design and manufacturing workflow. *Harvard Data Science Review*, Special Issue 5.

Martin, M. K.; Gonzalez, C.; and Lebiere, C. 2004. Learning to make decisions in dynamic environments: ACT-R plays the Beer Game. In *Proceedings of the Sixth International Conference on Cognitive Modeling*, 178–183. Pittsburgh, PA: Carnegie Mellon University/University of Pittsburgh.

McCarthy, J. 2007. From here to human-level AI. *Artificial Intelligence*, 171(18): 1174–1182.

Nelson, T. O.; and Narens, L. 1994. Why investigate metacognition? In Metcalfe, J.; and Shimamura, P., eds., *Metacognition: Knowing about knowing*, 1–25. Cambridge, MA: MIT Press.

Newell, A. 1994. *Unified theories of cognition*. Harvard University Press.

Oltramari, A.; Francis, J.; Ilievski, F.; Ma, K.; and Mirzaee, R. 2021. Generalizable neuro-symbolic systems for commonsense question answering. In *Neuro-Symbolic Artificial Intelligence: The State of the Art*, 294–310. IOS Press.

Panickssery, N.; Gabrieli, N.; Schulz, J.; Tong, M.; Hubinger, E.; and Turner, A. M. 2023. Steering LLaMA 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*.

Rahani, A. R.; and Al-Ashraf, M. 2012. Production flow analysis through value stream mapping: A lean manufacturing process case study. *Procedia Engineering*, 41: 1727–1734.

Raschka, S. 2024. *Build a large language model (from scratch)*. Simon and Schuster.

Ritter, F. E.; Schoelles, M. J.; Quigley, K. S.; and Klein, L. C. 2011. Determining the number of model runs: Treating cognitive models as theories by not sampling their behavior. In *Human-in-the-loop simulations: Methods and practice*, 97–116. London: Springer-Verlag.

Ritter, F. E.; Tehranchi, F.; and Oury, J. D. 2019. ACT-R: A cognitive architecture for modeling cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, 10(3): e1488.

Serrano-Ruiz, J. C.; Mula, J.; and Poler, R. 2021. Smart manufacturing scheduling: A literature review. *Journal of Manufacturing Systems*, 61: 265–287.

Stocco, A.; Sibert, C.; Steine-Hanson, Z.; Koh, N.; Laird, J. E.; Lebiere, C. J.; and Rosenbloom, P. 2021. Analysis of the human connectome data supports the notion of a "common model of cognition" for human and human-like intelligence across domains. *NeuroImage*, 235: 118035.

Subramani, N.; Suresh, N.; and Peters, M. E. 2022. Extracting latent steering vectors from pretrained language models. *arXiv preprint arXiv:2205.05124*.

Sumers, T. R.; Yao, S.; Narasimhan, K.; and Griffiths, T. L. 2024. Cognitive architectures for language agents. *Transactions on Machine Learning Research*, no volume.

Sutton, R. S.; and Barto, A. G. 1999. Reinforcement learning. *Journal of Cognitive Neuroscience*, 11(1): 126–134.

Swati, Z. N. K.; Zhao, Q.; Kabir, M.; Ali, F.; Ali, Z.; Ahmed, S.; and Lu, J. 2019. Brain tumor classification for MR images using transfer learning and fine-tuning. *Computerized Medical Imaging and Graphics*, 75: 34–46.

Turner, A. M.; Thiergart, L.; Udell, D.; Leech, G.; Mini, U.; and MacDiarmid, M. 2023. Activation Addition: Steering Language Models Without Optimization. *CoRR*, abs/2308.10248.

Ulrich, K.; Sartorius, D.; Pearson, S.; and Jakiela, M. 1993. Including the value of time in design-for-manufacturing decision making. *Management Science*, 39(4): 429–447.

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 31st International Conference on Neural Information Processing Systems*, NIPS'17, 6000–6010. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781510860964.

Wang, J.; Lan, C.; Liu, C.; Ouyang, Y.; Qin, T.; Lu, W.; and Philip, S. Y. 2022. Generalizing to unseen domains: a survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 35(8): 8052–8072.

Wu, S.; Oltramari, A.; and Ritter, F. E. 2024. VSM-ACT-R: Toward using cognitive architecture for manufacturing solutions. In *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, 69–79. Springer Nature.

Wu, S.; Souza, R. F.; Ritter, F. E.; and Lima Jr, W. L. 2023. Comparing LLMs for prompt-enhanced ACT-R and Soar model development: A case study in cognitive simulation. In *Proceedings of the 38th Annual Association for the Advancement of Artificial Intelligence (AAAI) Conference on Artificial Intelligence, Fall Symposium Series on Integrating Cognitive Architecture and Generative Models*, volume 2, 422–427.

Wulff, D. U.; and Mata, R. 2023. Automated jingle-jangle detection: Using embeddings to tackle taxonomic incommensurability. *OSF Preprints*.

Xu, L.; and Wang, W. 2023. Improving aspect-based sentiment analysis with contrastive learning. *Natural Language Processing Journal*, 3: 100009.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023. REACT: Synergizing reasoning and acting in language models. *International Conference on Learning Representations (ICLR)*.

Zhong, R. Y.; Xu, X.; Klotz, E.; and Newman, S. T. 2017. Intelligent manufacturing in the context of industry 4.0: A review. *Engineering*, 3(5): 616–630.

Zhu, F.; and Simmons, R. 2024. Bootstrapping cognitive agents with a large language model. In *Proceedings of the 2024 AAAI Conference on Artificial Intelligence*, volume 38, 655–663.

Zou, A.; Phan, L.; Chen, S.; Campbell, J.; Guo, P.; Ren, R.; and Hendrycks, D. 2023. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*.