
Position: We Need An Algorithmic Understanding of Generative AI

Oliver Eberle^{1,2} Thomas McGee³ Hamza Giaffar⁴ Taylor Webb⁵ Ida Momennejad⁵

Abstract

What algorithms do LLMs actually learn and use to solve problems? Studies addressing this question are sparse, as research priorities are focused on improving performance through scale, leaving a theoretical and empirical gap in understanding emergent algorithms. This position paper proposes *AlgEval*: a framework for systematic research into the algorithms that LLMs learn and use. AlgEval aims to uncover algorithmic primitives, reflected in latent representations, attention, and inference-time compute, and their algorithmic composition to solve task-specific problems. We highlight potential methodological paths and a case study toward this goal, focusing on emergent search algorithms. Our case study illustrates both the formation of top-down hypotheses about candidate algorithms, and bottom-up tests of these hypotheses via circuit-level analysis of attention patterns and hidden states. The rigorous, systematic evaluation of how LLMs actually solve tasks provides an alternative to resource-intensive scaling, reorienting the field toward a principled understanding of underlying computations. Such *algorithmic explanations* offer a pathway to human-understandable interpretability, enabling comprehension of the model’s internal reasoning performance measures. This can in turn lead to more sample-efficient methods for training and improving performance, as well as novel architectures for end-to-end and multi-agent systems.

1. Introduction

Large language models (LLMs) have soared to prominence, yet a fundamental question remains: What algorithms do LLMs actually use to solve problems? As the “gold rush” of scaling prioritizes practical breakthroughs, research priorities have centered on improving performance through scale, often regardless of guarantees or costs, while interpretability efforts have largely focused on understanding isolated mechanisms. Algorithmic understanding is often left behind. **This position paper argues that the ML community should prioritize research on an *algorithmic* understanding of generative AI.**

Existing work on understanding algorithmic operations in LLMs (Zhou et al., 2024; Li et al., 2023; von Oswald et al., 2024; Yang et al., 2024), while impressive, remain surprisingly few in number. Recent interpretability research has prioritized the exploratory analysis of low-level circuit mechanisms (Olah et al., 2020; Olsson et al., 2022), often without clear hypotheses, and even position papers that note the importance of algorithmic understanding only mention it as one among many other directions (Vilas et al., 2024). Algorithmic research on LLMs has taken a backseat among priorities, with theoretical work on the topic almost entirely lacking for both individual and multi-agent LLM systems. While multi-agent systems are now common solutions for reasoning and planning with Transformers (Wu et al., 2023a; Webb et al., 2024; Nisioti et al., 2024), theoretical foundations for efficiently building them remain underexplored.

While scaling has led to impressive results on a wide range of tasks, its limits remain unclear. Scale, rather than hypothesis-driven methods, has become the prevailing drive of general-purpose architectures since the rise of deep learning in the 2010s. This *Bitter Lesson* (Sutton, 2019), combined with the hypothesis that reward may be enough for the emergence of intelligence (Silver et al., 2021), have led to an emphasis on data- and compute-heavy approaches, prioritizing training data and fine-tuning with existing architectures. This trend has also widened the gap between frontier models and interpretability research, severely limiting their transparency, trustworthiness, and compliance with AI regulations (Samek et al., 2021; Kaur et al., 2022).

The field is increasingly encountering the limitations of available data and its quality, and the rising computational

¹Technische Universität Berlin, Berlin, Germany ²BIFOLD–Berlin Institute for the Foundations of Learning and Data, Berlin, Germany ³University of California Los Angeles, Los Angeles, USA ⁴Halicioğlu Data Science Institute, University of California San Diego, San Diego, USA ⁵Microsoft Research NYC, New York, USA. Correspondence to: Ida Momennejad <idamo@microsoft.com>, Oliver Eberle <oliver.eberle@tu-berlin.de>.

costs and diminishing returns of scale (Villalobos et al., 2024). This is in contrast to biological intelligence and brains, which provide an existence proof for a far more data- and energy-efficient approach. Recent studies suggest that optimizing inference-time compute can be more beneficial than simply scaling parameters, prompting shifts from feed-forward parameter growth to inference-time compute (Snell et al., 2024). Thus, given the environmental costs, scaling without understanding is not a sustainable path forward, particularly for multi-agent AI, where insights into system interactions are increasingly crucial.

This position paper calls for prioritizing systematic research on the algorithmic understanding of generative AI. An algorithm is typically defined as a finite set of rules or operations for transforming inputs into outputs (Turing, 1936; Knuth, 1968), which can be combined to form efficient strategies to solve larger problems. Algorithmic explanations of LLMs, therefore, involves uncovering the specific step-by-step procedures or “computational primitives” these models effectively learn and execute during task solving. By examining operations within a model’s architecture, its parameters, and inference process, we can comprehensively determine how the model arrives at its outputs.

A systematic framework for algorithmic understanding should address: a) What algorithms can generative AI learn, and how does this depend on factors such as model size, training data, fine-tuning, and in-context learning? b) Are there provable guarantees for any such algorithmic abilities? c) How can we build multi-agent systems in order to implement specific algorithms? d) How can we set algorithmic objectives for training and fine-tuning? e) How can we create a repository of algorithmic abilities? f) How can we study the selection and composition of these components to solve prompted tasks? g) How can we design architectures to guarantee specific algorithmic capacities? In what follows, we outline *AlgEval*, a research program for algorithmic evaluation and understanding of generative AI.

2. Related Work

To capture the rich internal computations implemented by increasingly complex machine learning (ML) systems, efforts in explainable AI and mechanistic interpretability have shifted focus to the inner workings of generative models, introducing approaches to uncover internal circuits (Olah et al., 2020; Wang et al., 2023), representations (Todd et al., 2024), dynamical motifs (Yang et al., 2024), and computational subgraphs (Schnake et al., 2022; Geiger et al., 2022), laying the foundation for an *algorithmic understanding* of model predictions.

Interpretability research initially emerged for deep classification models, before the era of generative AI (Lipton, 2017;

Montavon et al., 2018). Understanding classification models has largely focused on identifying relevant input features or heatmaps at intermediate layers, using methods such as perturbation-based (Lundberg & Lee, 2017), attention-based (Abnar & Zuidema, 2020), and gradient-based (Baehrens et al., 2010; Sundararajan et al., 2017; Ali et al., 2022) approaches. With the shift toward generative AI and sequence modeling, a principled understanding of internal processes, beyond input-output relations or isolated mechanisms, has become essential. This presents a significant technical challenge due to the scale and complexity of today’s frontier models. Thus, some researchers have focused on smaller or synthetic language models for targeted analysis and empirical studies, e.g., GPT-2 small (Wang et al., 2023; Conmy et al., 2023; Hanna et al., 2024) or toy Transformers (Liu et al., 2023; Ye et al., 2025).

On the other hand, early efforts to analyze LLM mechanisms focused on localizing specific functions within isolated model components (Vig & Belinkov, 2019; Clark et al., 2019), such as individual neurons (Gurnee & Tegmark, 2024; Zhou et al., 2018; Templeton et al., 2024), or individual attention heads (McDougall et al., 2024). More recent work has investigated how these components combine to form functional circuits (Olsson et al., 2022; Wang et al., 2023; Tigges et al., 2024), while other work has characterized the representations that support some higher-level computations, e.g., function vectors (Todd et al., 2024). Probing techniques have further been developed to assess whether specific properties can be accurately decoded from a model’s latent representations (Conneau et al., 2018; Hewitt & Manning, 2019), particularly in the analysis of reasoning strategies (Ye et al., 2025). While these approaches move toward a more integrated understanding, current findings are still largely fragmented, and we lack a solid theoretical foundation for understanding how these various components come together to implement algorithms.

3. AlgEval: Toward Algorithmic Evaluation and Understanding of LLMs

Algorithms consist of modular subroutines that exhibit compositionality, allowing them to be reused and recombined into efficient strategies for solving increasingly complex problems. From this conceptual starting point, AlgEval proposes a path toward algorithmic evaluation and understanding of LLMs through algorithmic primitives and their composition, analogous to a vocabulary and grammar. We then explore methods to evaluate them, from the common analysis of attention weights, latent representations, and circuit methods, to new approaches like inference-time compute and evaluating alternative solutions.

A key challenge in the algorithmic understanding of LLMs is designing tasks that are complex enough to support spe-

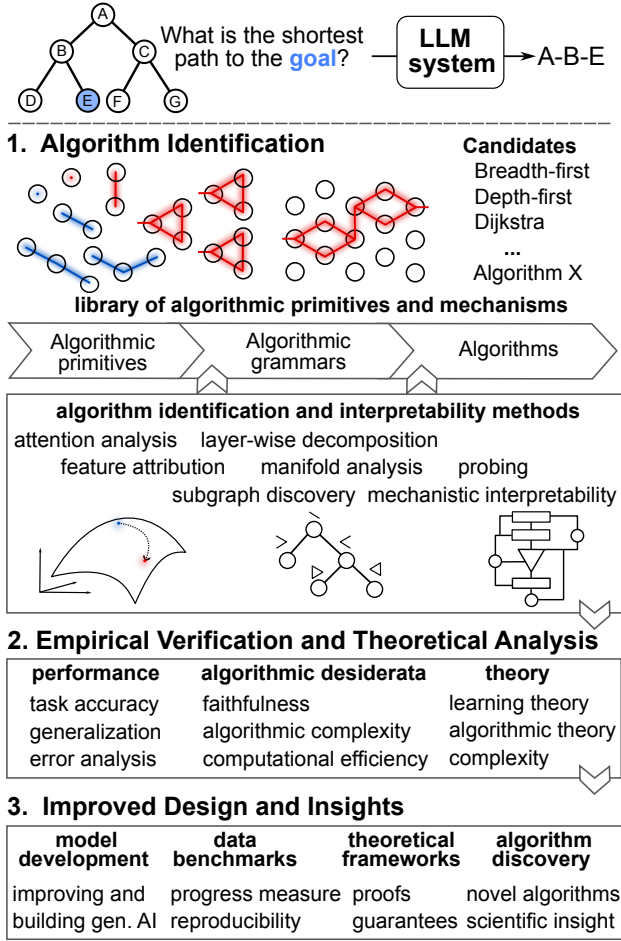


Figure 1. AlgEval: A methodological path to prioritizing algorithmic evaluation and understanding of LLMs.

cific algorithmic hypotheses and for which ground truth responses and strategies are available. An example is goal-directed navigation on deterministic graphs, Figure 1 (top). To solve such a task, classical search algorithms like breadth-first search (BFS), Dijkstra, or depth-first search (DFS) offer a verifiable algorithmic ground truth for evaluation. Moreover, heuristic, optimization-based approaches like simulated annealing (Kirkpatrick et al., 1983; Cerný, 1985; Metropolis et al., 1953), or amortized inference (Gershman, 2019) could be at play. For instance, in amortized inference or “learning to infer,” rather than solving problems from scratch, the model compresses frequently used inference routines into a parametric function. In AlgEval, these hypotheses guide model analysis by evaluating internal mechanisms and inference-time generation to extract functional structure.

This hypothesis-driven approach re-centers the understanding and development of LLMs on scientific principles, emphasizing rigor in evaluating learned mechanisms. Drawing from Marr’s three levels of analysis, i.e., computational, algorithmic, and implementation (Marr, 1982; Vilas et al.,

2024), AlgEval prioritizes understanding mechanisms at the algorithmic level over their physical realization. By systematically testing algorithmic hypotheses, we also move beyond the assessment of behavioral goals at the computational level to uncover internal structures that enable robust problem-solving. Next, we describe what we refer to as *algorithmic primitives*, and how models can piece them together to form algorithmic solutions.

3.1. Algorithmic Primitives and Vocabulary

Basic primitives. We define primitives to be the basic elements necessary to realize a specific algorithm. By iteratively breaking down an algorithm, a vocabulary of essential primitives can be obtained. In the context of LLMs, a variety of such primitives have been identified through methods from interpretability and data analysis. These approaches have led to the discovery of vector-based mechanisms, including function vectors for in-context learning mappings (Todd et al., 2024; Edelman et al., 2024), vector arithmetic computations (Merullo et al., 2024), steering vectors (Yang et al., 2024), copy suppression (McDougall et al., 2024), and key-value memory retrieval (Geva et al., 2021). Additional primitives have been described that identify duplicate elements, and inhibit or increase attention to specific sequence elements (Wang et al., 2023).

The functions of algorithmic primitives, as we defined them, should be predominantly domain-general and applicable to a wide range of sequence modification tasks. While the interpretability community provides a starting point to discover possibly domain general primitives, e.g., in-context learning (ICL) (Elhage et al., 2021; Olsson et al., 2022), token binding (Vasileiou & Eberle, 2024; Feng & Steinhart, 2024), or recognizing semantic relationships (Ren et al., 2024), principled frameworks for connecting and combining these individual findings are mostly lacking. On the other hand, inspired by the Transformer architecture, a framework of programming primitives, termed ‘restricted access sequence processing language’ (RASP) (Weiss et al., 2021; Zhou et al., 2024), has been developed to formalize the study of algorithmic implementations in Transformers’ interpretable sequence operations, including basic select and aggregate manipulations. While helpful, RASP focuses on what could be built with LLMs in minimal scenarios, currently remaining inapplicable to understanding real world generative AI.

A key goal of AlgEval is the identification and evaluation of algorithmic primitives as clearly defined operations that can be assessed and validated at a lower level of complexity, and bridged to more sophisticated algorithmic levels. Similar to current discussions on the emergence of universal representations (Huh et al., 2024), this enables building a systematic catalog of universal primitives across models and tasks.

Primitives as foundational algorithms. Essential primitives can be composed into domain-general basic algorithms for sequence generation tasks, leading to the discovery of algorithmic subgraphs and circuits that integrate multiple primitives and mechanisms. A number of disjointed findings are noteworthy. For instance, a network of induction, inhibition/excitation, duplicate token, and copy heads have been identified to solve the task of indirect object identification (Wang et al., 2023). Different algorithms have been discovered that solve modular arithmetic via combinations of circular embeddings and simple trigonometric operations (Nanda et al., 2023; Zhong et al., 2023), which can be understood through different analytical solution approaches. By combining memory heads, that promote internally stored information, with in-context heads, a mechanism for factual recall has been reconstructed (Yu et al., 2023). We consider these efforts as starting points for studying simple combinations of primitives. On the other hand, a set of theoretical studies focusing on complexity analysis (Elmoznino et al., 2024), algorithmic selection and assembly theory (Sharma et al., 2023), and combining and analyzing primitives via structured interactions (Morris et al., 2019; Eberle et al., 2022; Schnake et al., 2022; Fumagalli et al., 2023), call for an integration of theoretical and empirical contributions toward algorithmic discovery.

Algorithms as primitives. Recent studies have demonstrated how Transformers can display classic ML methods, including kernel-based approaches (Tsai et al., 2019), support vector machines (Tarzanagh et al., 2023), Markov chains (Zekri et al., 2024), higher-order optimization methods for ICL (Fu et al., 2024), and temporal difference learning (Demircan et al., 2024). Further research is required to understand whether they can be understood as algorithmic primitives that can be combined, further broken down into more basic primitives, or both. This highlights the need to understand the extent to which primitives serve as task-specific or domain-general building blocks, and whether we can identify a systematic hierarchy of primitives.

3.2. Algorithmic Composition

Compositionality. The combination of algorithmic primitives into more complex algorithms is a form of *compositionality* that, in principle, should support the construction of a vast number of combinations from a finite vocabulary. The search for algorithms is thus related to the broader debate about the extent to which LLMs are capable of compositional reasoning and generalization. While there is some evidence to support the existence of compositional representations and mechanisms in generative models (Lepori et al., 2023; Campbell et al., 2024), there is also evidence for persistent failures in tasks that require compositional reasoning (Lewis et al., 2022; Mitchell et al., 2023; Conwell et al., 2024). Studying the algorithms used by these models

can clarify whether their reasoning is truly compositional or reflects ineffective strategies like memorization (Power et al., 2022; Qiu et al., 2022), or instabilities in representing rare events (Kandpal et al., 2023).

An algorithm’s relevant primitives can be combined through several strategies. Previous works have explored optimization-based methods, such as learned graph structures and message passing (Geiger et al., 2022), that enable dynamic interaction among components, while symbolic and compositional approaches provide explicit modular frameworks for task-specific solutions (Geiger et al., 2022; Wu et al., 2023b). ICL techniques, like providing grammar through input (Galke et al., 2024), can steer model behavior based on contextual cues. In the context of RASP, hard-coded aggregate functions can be used to nest several primitives, forming more complex functions, e.g., able to reverse or sort sequences (Weiss et al., 2021). So far, it remains unclear whether compositionality emerges when learning next-token prediction on, for example, a trillion tokens of compositional data, and to what extent it can be built into the architecture or training methods.

To foster deeper understanding of algorithmic composition, we need targeted empirical studies combined with building compositionality from first principles. Promising starting points include imposing known compositional constraints on the grammar, e.g., via hierarchical pyramids (Lin et al., 2017), equivariant structure (Satorras et al., 2021), conditions imposed by the data-generating function (Wiedemer et al., 2023), or algorithmic complexity (Elmoznino et al., 2024). Assembly theory offers an evolutionary perspective on forming complex algorithmic grammars by selecting and recombining primitives (Sharma et al., 2023).

3.3. Methodologies for Algorithmic Evaluation

AlgEval focuses on methodological advances necessary to identify, evaluate, and discover algorithms. This motivates combining existing interpretability techniques with novel approaches to capture the complexity of modern AI.

Analyzing representations and attention. Neural representations, patterns of activity across neural populations or intermediate Transformer layers, are increasingly recognized as key to understanding or modifying network computations (Zou et al., 2023; Sucholutsky et al., 2024). Analyses of representational similarities within or across layers, using an array of similarity measures sometimes inspired by cognitive science and neuroscience, elucidate how these layers transform information (Kornblith et al., 2019; Klabunde et al., 2024; Yousefi et al., 2024; Sucholutsky et al., 2024; Williams et al., 2021; Giaffar et al., 2024). For instance, LLM embeddings can reveal interpretable structures for deception detection by identifying a 2D subspace encoding true/false statements (Bürger et al., 2024), and a three-stage

process of deceptive behavior has been uncovered through low-dimensional projections (Yang et al., 2024).

Complementary to representation analyses, attention in LLMs is commonly analyzed to identify message passing operations among tokens. Layer-wise attention scores help interpret token importance and internal model structures, including attention rollout and attention flow (Abnar & Zuidema, 2020). To capture task-specific model processing, feature attribution methods compute feature importance scores, addressing the limits of attention analysis in providing faithful explanations (Wiegrefe & Pinter, 2019), with techniques like saliency (Chefer et al., 2021) and modified gradient methods (Ali et al., 2022; Achibat et al., 2024; Jafari et al., 2024). Furthermore, internal analysis of attributions can aid in discovering relevant representational concepts (Kauffmann et al., 2022; Chormai et al., 2024).

The integration of attention and representation analyses is a key feature of AlgEval, as information passing between tokens and representation analyses can uncover network structures and transformations, helping us understand algorithmic primitives and compositionality in problem-solving. (see Section 4).

Subgraphs and circuits. The identification of relevant internal structure presents a current methodological frontier in our understanding of LLMs. Various methods have been proposed to extract circuits (Olah et al., 2020; Wang et al., 2023), subgraphs (Schnake et al., 2022; Geiger et al., 2022), feature interactions (Eberle et al., 2022; Fumagalli et al., 2023; Vasileiou & Eberle, 2024; Kauffmann et al., 2024), and causal symbolic models (Geiger et al., 2022). Key techniques for discovery of internal structure include activation patching (Wang et al., 2023), automatic circuit discovery (Conmy et al., 2023), attribution patching (Syed et al., 2024; Hanna et al., 2024), and graph explanations (Schnake et al., 2022; Sanford et al., 2024). Viewing LLMs as graphs provides a complementary perspective of sequence processing as computations that extend across multi-hop neighborhoods that form substructures (Besta et al., 2024), build motifs and compute higher-order interactions across neurons (Eberle et al., 2022; Schnake et al., 2022; Fumagalli et al., 2023).

3.4. Identifying and Structuring Primitives

Methodologically, we have identified the following five steps as crucial components of AlgEval: (a) **Identify and form a library of primitives** which can grow over time and anchor corresponding tasks and mechanisms. Each algorithmic primitive can correspond to multiple tasks and algorithms, supporting different mechanistic implementations. Primitives can be either hypothesis-based, rooted in decades of theoretical algorithm research, or empirically observed. (b) **Build a collection of simple tasks** which require a set of primitives for their solution. Examples in-

clude sequence induction (Olsson et al., 2022), or copying a sequence of unique tokens (Zhou et al., 2024). (c) **Create a library of mechanisms** that implement primitives, along with corresponding interpretability and analysis tools to identify them as discussed in Section 3.3. (d) **Analysis of Composition** is crucial for identifying primitives, as it involves understanding how they combine, how algorithms are implemented across layers and inference, and whether compositional patterns generalize across tasks and models. (e) **Ablations** serve to identify and evaluate primitives’ role, which necessitates developing tools for causal intervention and ablation (Geiger et al., 2022; Talon et al., 2024), as well as using statistical tests. Newly discovered primitives, along with their associated tasks, mechanisms, and methods, are then added to the growing library of primitives for future analyses and integration into new models.

3.5. New Directions for Algorithmic Analysis

The role of in-context learning. Recent work has explored how ICL improves transformer performance (Fu et al., 2024), one proposing Transformers are algorithms (Li et al., 2023). A recent paper analyzed changes in attention and representation due to ICL and how it related to improvements in behavior (Yousefi et al., 2024). While that work did not focus on interpreting the effect of these changes on algorithmic primitives or composition, one important future direction will be to analyze the algorithmic consequences of ICL, such as promoting the use of specific algorithms.

Inference-time compute. Inference-time compute has recently arisen as a new paradigm for reasoning with LLMs. In this approach, rather than solving a problem via a single feedforward pass, the autoregressive outputs of the model can be used to perform intermediate computations. Examples of this approach include chain-of-thought (Wei et al., 2022), explicit tree search (Yao et al., 2024), agent-based approaches (Webb et al., 2024), and models such as o1 (Jaech et al., 2024) and the open-source R1 (DeepSeek-AI et al., 2025) that are trained to perform inference-time compute via amortized optimization. Some have even argued for scaling inference-time compute instead of scaling parameters or training (Snell et al., 2024).

AlgEval also applies to emergent algorithms at inference time, where sequential outputs can be more amenable to algorithmic analysis than high-dimensional feedforward computations. For example, LLMs can be trained to explicitly search via their outputs, implementing search procedures like exploration and backtracking through traces provided in context (see *in-context search*, *SearchFormer*, *Stream of Search*) (Gandhi et al., 2024; Lehnert et al., 2024). This is particularly revealing when models acquire algorithms through amortized inference for downstream tasks rather than via supervised fine-tuning or ICL, potentially yielding

novel or emergent solutions.

While no existing work systematically examines inference-time emergent algorithms, likely due to their novelty, the core AlgEval principles of identifying algorithmic primitives, forming top-down hypotheses, and bottom-up testing remain applicable. There may also be interactions between feedforward algorithms and inference-time compute, where certain processes like search are offloaded to the output space, allowing the feedforward pass to specialize in different primitives. Finally, it is crucial to ensure causal linkage between inference-time outputs and actual performance, since chain-of-thought can be unfaithful or unrelated to the model’s real reasoning (Turpin et al., 2024; Stechly et al., 2024).

Reinforcement learning and memory compression. A recent open-source LLM, DeepSeek R1 (DeepSeek-AI et al., 2025), used reinforcement learning (RL) and inference-time compute to match the performance of OpenAI’s o1 at a fraction of the training cost. Interestingly, this model displayed an apparently emergent form of backtracking (referred to as an ‘aha moment’). A future direction of research is to study to what extent this behavior emerged purely as a consequence of training with RL, as opposed to relying on documents in the base model’s training data that include similar examples of backtracking (annotated math solutions). Given o1 was supposedly also trained on annotated math solutions, it is important to study how training with RL, particularly Group Relative Policy Optimization (GRPO) (Shao et al., 2024), is necessary to elicit this behavior. Open-source models like DeepSeek R1 offer an opportunity to study how training data and RL shape their algorithmic vocabulary and compositions thereof. A key question herein is whether RL led to cached strategies for amortized inference, enabling “learning to infer,” by compressing prior inferences (Gershman, 2019; Radev et al., 2020). Such strategies can link learning to compressed memory representations, which is also observed in the human brain and behavior (Momennejad et al., 2017; Russek et al., 2017; Momennejad, 2020; Brunec & Momennejad, 2021; Momennejad, forthcoming).

4. Case Study

To ground our position in an empirical example, we conducted a case study focused on LLMs, which have been shown to perform poorly on graph navigation and multi-step planning tasks (Momennejad et al., 2023). In cases where they do succeed, it remains unclear how they solve these problems, e.g., whether they implement classic search algorithms or use other strategies. To address this question, we studied the algorithms used by widely used LLMs, instruction-tuned Llama-3.1 with 8B and 70B parameters, in the context of graph navigation. We considered a simple tree graph structure, presented in a prompt that describes

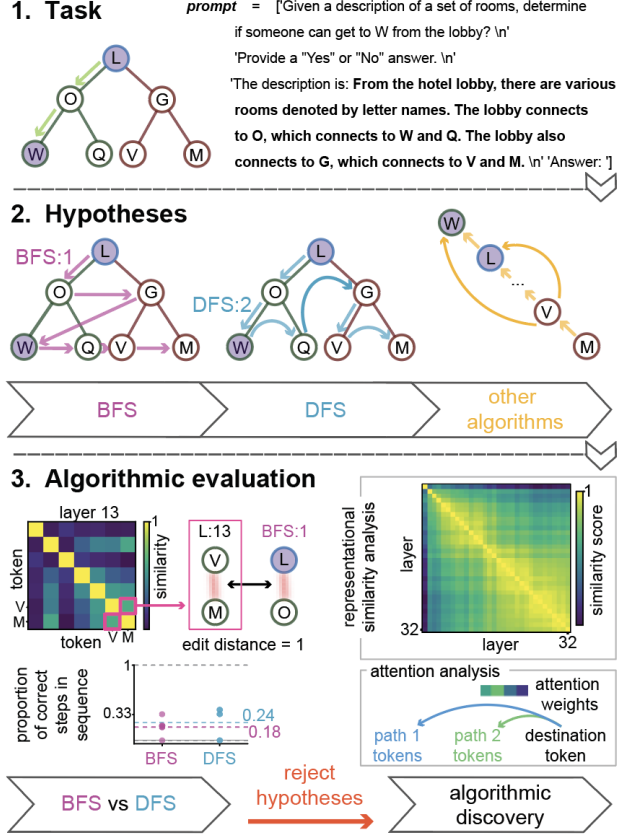


Figure 2. Case study. A graph navigation task, algorithmic hypotheses (possible rollouts for DFS and BFS are shown), and potential methods for algorithmic evaluation.

how rooms (nodes) connect to one another (edges) and tasks the model with determining whether a direct path from the start to the goal node exists (Figure 2).

A straightforward hypothesis to test is that the model may use standard search algorithms like DFS, BFS, or Dijkstra to find paths between graph nodes, with each layer potentially representing one search step. This analysis assumes that each layer corresponds to one visited node or node pair and that multiple connections may be evaluated simultaneously. If the layers successfully identify the correct path, we can infer its sequence of visited nodes by examining which node tokens receive the highest attention or exhibit the strongest representational similarity across layers (Kriegeskorte et al., 2008; Kornblith et al., 2019; Manvi et al., 2024). These combined analyses of attention and representation provide insight into the potential step-by-step algorithmic procedures underlying the LLM’s graph exploration.

Prompt. We introduce the model to a two-step tree graph following the prompt from Momennejad et al. (2023), which demonstrated that LLMs struggle with graph navigation and especially tree search. The model is tasked with determining the validity of a given path, producing a single token output: ‘yes’ or ‘no’. The full prompt and task for starting from

the ‘lobby’ and goal location W are shown in Figure 3a. Note that some nodes appear multiple times in the prompt (e.g., ‘lobby’ is repeated four times), thus in our analysis in Figure 3c, we display the representation trajectory for each appearance of a node.

4.1. Cascading Attention Analysis

Analyzing layer-wise attention matrices in LLMs (Vaswani, 2017) provides a direct and commonly used way to trace message-passing operations among tokens. To better understand how the model’s algorithm leverages attention, we analyzed 1) the attention from each graph node to its preceding nodes and the goal, and 2) the attention from the final token to all nodes. The former allowed us to test how the graph nodes “search over themselves” across layers, while the latter allowed us to test how the model searches over relevant tokens at inference time. We next present results on Llama-3.1-8B with additional analyses of the 70B model presented in Appendix A.4.

To test whether the model performs search over graph nodes across layers, we analyzed how the final token’s attention shifts between correct and incorrect paths as shown in Figure 3. We first used a linear mixed-effects model with logit-transformed average attention as the outcome variable, pathway (correct or incorrect) as a fixed effect, and layer number as a random intercept. Results indicated that the final token allocated significantly more attention to rooms in the correct pathway than to those in the incorrect one ($b = 0.33$, $SE = 0.07$, $t(2015) = 4.51$, $p < .001$) (Appendix Table 1). The model included a random intercept for layer number (variance = 0.96, $SD = 0.98$), and the residual variance was 2.80 ($SD = 1.67$). A layer-wise analysis showed that the final token allocated significantly more attention to rooms on the correct path in 14 of 32 layers (Appendix Table 2), with only three layers exhibiting significantly more attention to the incorrect path (Appendix Table 3).

Analyzing attention to individual nodes, response tokens, and the goal token revealed an interpretable layer-by-layer sequence leading to the correct response: early-to-mid layers attended to pairwise node links, while attention to the goal node W peaked in layers 13–14, and the final token’s attention shifted to the correct response around layer 19.

The cascading attentional spread from each node to its predecessors may incrementally direct attention to the correct path, suggesting a **policy-dependent** algorithm rather than an exhaustive search like BFS or DFS. The model seems to (1) incrementally attend to the path leading to the goal via query-key attention weights, then (2) attend to the goal token in later layers as presented in Figure 3.

4.2. Analysis of Feedforward Representations

To characterize how graph representations contained in feed-forward activity change across LLM layers, we first defined the token-by-token representational similarity matrices V^i for each layer, indexed i , entries of which $V_{xy}^i = u_{i,x}^T u_{i,y}$ are inner products between activation vectors $u_{i,x}$ and $u_{i,y}$, for room tokens x and y respectively. To ask if we could identify discrete changes in the graph representational geometry between neighboring layers that might be interpreted as steps of an algorithm, we computed layer-by-layer similarity matrices, S^d using similarity measure d (Kornblith et al., 2019; Williams et al., 2021), where $S_{ij}^d = d(V^i, V^j)$. From layers 4 to 32, the representational similarity between neighbouring layers remains high (with scores above 0.95), suggesting that graph representational geometry changes relatively smoothly from early to late layers and does not appear to change in a clear step-like fashion as shown in Appendix Figure 6.

Comparing LLM vs. hypothesis sequences. To compare the model’s feedforward activations to classical search algorithms, we extract a possible sequence of algorithmic steps from the LLM hidden unit activations: for each layer i we identified the node pair $e_i = (x, y)$ with highest representational similarity in V_i (Figure. 2). We compared this sequence of edges with all possible unique rollouts of BFS and DFS by: i) computing the edit distance between each LLM step and each BFS/DFS trajectory, and ii) finding the longest sequence of correct steps for each trajectory; to be maximally permissive, we only required LLM steps to be in the correct sequence, not necessarily in adjacent layers. This analysis revealed that the sequence of steps identified in LLM layers is not well matched to any full trajectory under either BFS or DFS (the mean proportion of correct matching steps in sequence is 0.18 for BFS and 0.24 for DFS; Figure 2).

Competing representations. We next analyzed the evolution of node representations across layers. In Figure 3c, we present a low dimensional latent space (t-SNE) projection of node representations across all layers, along with the final end-of-sequence (‘eos’) representation (‘yes’ or ‘no’). Colors represent room appearances (graph nodes) in the prompt, and numbers indicate the layer from which the representation is extracted, showing the evolution of representational distance among rooms over layers.

While in the first layers all nodes are closely clustered together, across the layers, we observe a clear progressive separation of tokens associated with the ‘lobby’ from other nodes (Figure 3c). This hints at an algorithmic strategy that distinguishes between the anchor tokens (the ‘lobby’) and a set of varying potential goal nodes. Interestingly, we find further structure developing across the latter graph nodes: a consistent clustering of non-goal nodes, and a grouping of

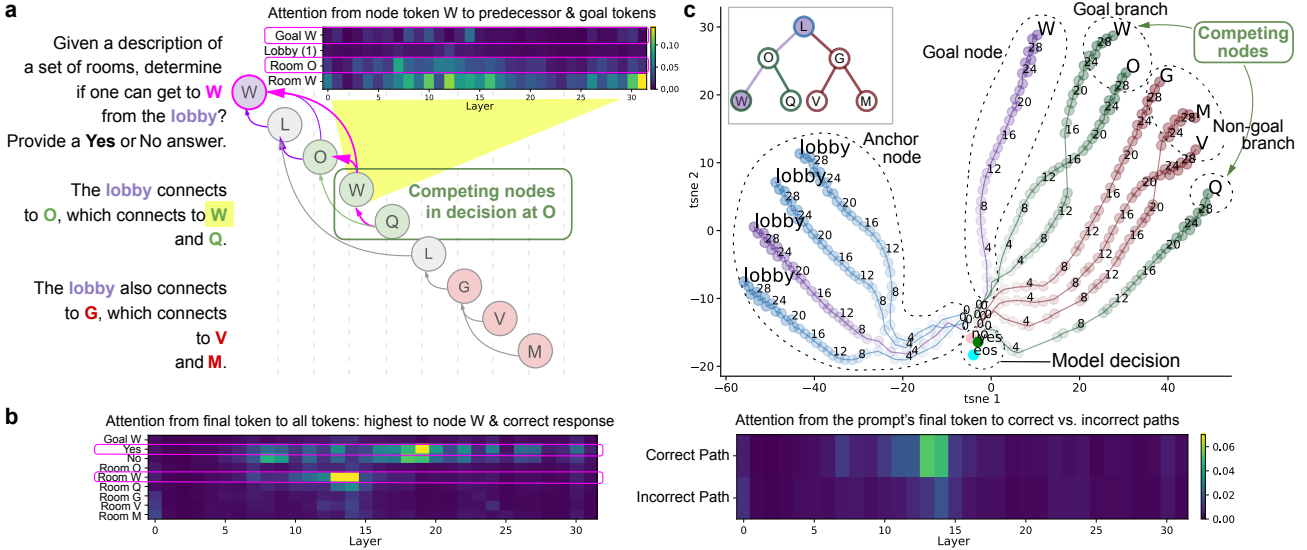


Figure 3. Case study on algorithmic discovery. (a) Attention heatmap: from goal to the correct vs. incorrect pathway. (b) Attention is mostly allocated to the goal location M, followed by V and G on the correct path. (c) Separation of room representations across layers.

subsets among them as shown in Appendix Figure 5. Similar to our attention analysis results (Figure 3a), we observe that the competition between the goal node W and Q, W’s closest competitor node in the same branch, is reflected in their progressively increasing separation in representation space (see Figure 3c and Appendix Figure 5). This separation typically emerges at intermediate layers and follows the high attention that W receives from the final ‘eos’ token at these stages (Figure 3b). Lastly, we observe that the representation of the goal token, defined in the prompt context, consistently maps onto a distinct trajectory, separating its representation from all other nodes while being closest to the ‘lobby’ representation (purple goal branch in Figure 3c).

Search strategies in larger models. To test whether increased scale results in a different, potentially more robust algorithmic search strategy, we repeated our experiments on a ten times larger model (Llama-3.1-70B-Instruct). As shown in Figure 7 and Figure 9 in the Appendix, our results are consistent to those of the smaller model, displaying similar patterns in both representation space and attention scores, but with a more pronounced separation between correct and incorrect trajectories. We further compared the sequence of states with the highest representation activation across layers to potential search sequences generated by BFS and DFS strategies, and found no differences in search strategy between the models as shown in Figure 8 in the Appendix. Neither sequence of representations closely matched the BFS or DFS rollouts on the target graph.

Interpretation. Using the framework of AlgEval, we found that attentional and representational patterns in LLMs do not align neatly with classical search algorithms, highlighting the role of algorithmic explanations in making such

discrepancies interpretable. The attention patterns we observed (Figure 3b and Figure 9 in the Appendix) suggest that current models do not construct a full world model or perform exhaustive search. We further find that representations of graph nodes evolve layer by layer, increasing the representational distance among key nodes, e.g., the distance between the closest competing node and the goal node (W vs. Q, Figure 3 and Figure 7). Future studies should verify whether this competition-driven separation across layers, also observed in simpler settings (Wang et al., 2023), represents an algorithmic primitive, and whether it is driven by inhibition or mover attention heads, function vectors (Todd et al., 2024), or other mechanisms. Furthermore, it remains to be seen to what extent a model’s search strategy varies with task and model complexity, especially in relation to its failure modes when navigating more complex graphs.

5. Alternative Views

Our proposal has some overlap with mechanistic interpretability (Olah et al., 2020; Olsson et al., 2022), but differs with key distinctions. First, mechanistic interpretability often emphasizes bottom-up perspectives, even advocating for hypothesis-free circuit analysis (Olah, 2023). Second, while it is good to remain open-minded about hypotheses, it has been argued that interpreting data with an ‘innocent eye’ is not possible (Gershman, 2021). Thus, we advocate for combining top-down algorithmic hypotheses with bottom-up evaluation. Third, while mechanistic interpretability focuses on low-level circuits, AlgEval targets the algorithmic level of explanation (Marr, 1982), integrating primitives and their composition, from circuits to higher-level computations. This perspective also connects to work on neural

algorithmic reasoning (Veličković & Blundell, 2021), which aims to integrate algorithmic structure into neural architectures by operating in high-dimensional latent spaces while performing computations aligned with a specific target algorithm. Fourth, we contrast our approach with “AI-assisted interpretability,” in which AI systems explain other AI systems (Choi et al., 2024; Li et al., 2024a; Olah, 2023). Although it has produced intriguing results, like automated neuron descriptions (Choi et al., 2024), we maintain it cannot replace hypothesis-driven research. Designing rigorous algorithmic tests likely requires reasoning skills beyond current models, so human involvement remains vital. Finally, AlgEval may offer an alternative to the dominant scaling paradigm (Sutton, 2019) by systematically understanding emergent algorithms and embedding them into architectures, rather than relying solely on training data and scale.

6. Discussion and Future Directions

In this position paper, we advocate for systematic research into algorithms learned and used by generative AI. We introduced AlgEval as a framework to investigate algorithmic primitives, their composition, and the impact of architecture, parameters, and optimization. AlgEval extends to inference-time compute and how it depends on training data and objectives, with implications for both empirical design and theoretical understanding.

Theoretical research. Recent theoretical work has addressed hierarchical language learning in Transformers (Allen-Zhu & Li, 2024), what formal languages they can learn (Strobl et al., 2024), their representational strengths and limitations (Sanford et al., 2023), how they learn shortcuts to automata (Liu et al., 2023), and how chain-of-thought provably improves their computation (Malach, 2023; Li et al., 2024b). However, this work remains disjointed from interpretability research, and approaches to mechanistic understanding often lack formal theory. This makes connecting high-level explanations to low-level processes and generalizing insights across architectures difficult. Meanwhile, multi-agent systems show empirical benefits but remain theoretically under-explored, particularly regarding how agents interact, which algorithms they perform, and how their strategies differ from end-to-end models. Overall, theoretical work on the algorithmic capacities of LLMs is still sparse. Future studies should investigate properties of good solutions (axiomatic desiderata), optimality and complexity proofs, and learning theory for LLM algorithms. A key question is whether certain architectures, parameters, and optimization schemes can guarantee the implementation of specific algorithms.

Algorithm evaluation should consider a number of relevant desiderata: *Fidelity*, which ensures consistent input–output reliability, *optimality*, which pursues the most efficient solu-

tions (Shneidman & Parkes, 2004), and *minimality*, which values lower algorithmic complexity (Elmoznino et al., 2024). Further, algorithms should be *expressive* enough to be clearly understood and *runtime efficient* for scalability (Swartout & Moore, 1993). These properties form a basis for evaluating and developing algorithms in ML systems. An important direction is algorithm-centric architecture design and the reuse of verifiable algorithmic components, e.g., exploring symbolic operations to evoke specific algorithms. Another direction is to incentivize algorithmic building blocks during training, e.g., through complexity or sparsity regularization.

Steering, guiding, optimization. Optimizing a given LLM to learn or use specific algorithms can be steered via training, architecture, or ICL. Future work should examine how ICL can steer models toward a target algorithm and how feedforward or attention-based modifications improve primitives or composition. Another possibility is inference-time optimization to encourage specific algorithms. Ultimately, we can train or fine-tune LLMs to implement specific algorithms, guided by an algorithmic perspective.

Designing new architectures. A key future direction is algorithm-centric architecture design, which may be accomplished through the identification and re-use of verifiable algorithmic components. It would be especially interesting to see, for instance, if specific algorithms can be evoked through the use of symbolic operations, in context learning, or training regimes, e.g., GRPO (Shao et al., 2024) as opposed to Proximal Policy Optimization Algorithms or PPO (Schulman et al., 2017). An algorithmic understanding of architectural choices can improve how we build end-to-end architectures and multi-agent AI systems (Webb et al., 2024), paving the way for more transparent, reliable, and theoretically grounded generative AI. This enhanced algorithmic understanding can have significant implications for scientific applications of ML, potentially uncovering efficient strategies and fundamental mechanisms across various domain sciences.

Conclusion. While interpretability research has begun moving toward mechanistic and circuit-level analysis, it largely overlooks algorithmic explanations and evaluation of LLMs. **Our position is that the next generation of ML researchers should prioritize algorithmic understanding of generative AI.** We have presented AlgEval, approaches for algorithmic research, and actionable next steps. Deepening our understanding of how LLMs compute can enhance their sample efficiency, reduce emissions, and improve safety compliance, ultimately strengthening the overall impact of our community’s work.

Acknowledgements

The authors acknowledge Eder Sousa, Aishni Parab, Dalal Alharthi, Montassir Abbas, Andrea Kang, Hongjing Lu, Mark Green, Peter Todd, and the participants of the Institute for Pure and Applied Mathematics (IPAM) Long Program on the Mathematics of Intelligences. Part of this research was performed while some of the authors were visiting IPAM, which is supported by the National Science Foundation (Grant No. DMS-1925919).

Impact Statement

We propose prioritizing a systematic understanding of algorithmic primitives and compositions in LLMs, and how they are learned and used given their architecture parameters, and training data. Below are a number of near-term and long-term impacts that can further motivate this research priority. We believe this research will impact more than mere understanding, and could potentially lead toward more efficient and lower-emission training and improving generative AI, and designing new architectures with algorithms in mind.

The rapid success and rise of generative AI faces the challenges of unpredictable errors as well as large emissions. In spite of these known challenges, the ML community’s research priorities in the area of generative AI and LLMs have so far been myopically focused on improving performance, regardless of costs, and mechanistic interpretability, regardless of guarantees or algorithmic understanding. This position paper argues that the ML community should shift its focus to the algorithmic level of analysis, as the current priorities are often wasteful and offer only limited insight into models’ fundamental inner workings.

A previous position paper on inner interpretability (Vilas et al., 2024), inspired by the analogy to Marr’s levels in neuroscience (Marr, 1982), calls for attention to all levels: the computational level, the algorithmic level, and the implementation level. While we agree, we think there is a specific need to prioritize an algorithmic understanding in line with long-standing traditions in computer science. We highlight the importance of putting resources into understanding the algorithmic vocabulary and grammar of generative AI, which will contribute to the field beyond understanding isolated phenomena and toward a more systematic foundation.

Sample Efficiency and Improved Behavior

An algorithmic understanding of generative AI can empower us to identify methods for improving sample-efficiency. This could occur through improving training with algorithmic performance in mind, optimizing and scaling inference-time compute and chain of thoughts with an understanding of their algorithmic implications. The latter could in turn im-

prove the nebulous state of prompt engineering. Moreover, algorithm-based training and the reuse of algorithmic components in future models, both end-to-end and multi-agent, could lead to better behavioral performance and improved generalization. Ideally, this would yield fewer iterations for a given task, especially in multi-agent architectures. Together, sample efficiency during training and compute efficiency during response generation could potentially address both compute and emission challenges of generative AI.

Emission Efficiency

Many assume generative AI’s emission costs occur only during training, which equate to the lifetime emissions of multiple cars (Strubell et al., 2019). However, significant costs arise during use as well. For instance, a single interaction with a state-of-the-art LLM can require half a liter of water for cooling and emit the carbon equivalent of five gallons of gas when solving challenging problems with OpenAI’s GPT-3. It is estimated that by 2027, global AI’s water usage could rise to nearly two-thirds of the United Kingdom’s annual consumption (Li et al., 2025). Repeated errors and back-and-forth interactions further increase these water and carbon costs, potentially leading to catastrophic climate impacts when scaled globally. Adopting algorithm-driven approaches can enhance emission efficiency in training and reasoning, and inform the design of new architectures. We believe that an algorithmic understanding of LLMs can lead to more environmentally sustainable methods, which is crucial as these models are rapidly integrated into everyday products.

We believe an algorithmic understanding of LLMs can lead to less wasteful approaches, considering the environmental costs of training and using generative AI. This is especially important given the rapid pace at which these models are being integrated into everyday products.

Theories for Multi-Agent AI

Given the unreliability of most generative AI approaches, multi-agent LLM architectures have become common to correct LLM errors and hallucinations, orchestrate actions, and improve reasoning, among other use cases. While these approaches make LLM-based solutions more reliable, they lead to even higher emissions. On the other hand, various approaches are a patch-work of popular knowledge of psychology, and at best cognitive science or brain-inspired approaches, without any systematic theories or guarantees. One of the key challenges lies in building and analyzing agentic systems with provable performance, ensuring that they can function reliably and effectively in complex environments. We believe a better understanding of what LLMs truly do can lead to more efficient design of multi-agent systems as well as their implementation and product integration,

with advantages for engineers, users, and the planet.

Trust, Compliance, and Safety

An enhanced understanding of model behavior enables researchers to discover novel mechanisms and advance a principled understanding of generative AI. An algorithmic understanding of LLMs can, in turn, increase the interoperability and trustworthiness of models. These insights can empower researchers and engineers to ensure compliance with safety standards.

Algorithmic Bias

The challenge of bias in computer systems (Friedman & Nissenbaum, 1996), particularly instances of *technical bias* require an in-depth understanding of the underlying systems. In the context of generative AI, today’s systems are commonly found to make unfair, undesired and even harmful predictions (Shah et al., 2020; Lucy & Bamman, 2021; Eberle et al., 2023; Fang et al., 2024), raising concerns about their deployment in sensitive domains. AlgEval directly supports the detection and understanding of algorithmic bias by systematically evaluating and interpreting a model’s fundamental components. As bias can manifest at different levels within a system, for example, in low-level primitives, compositions thereof, or in the functioning of full algorithms, AlgEval offers a complementary approach to addressing pre-existing bias originating from training data. By targeting technical bias at these distinct levels, it enables a more granular and thorough evaluation of bias in generative models.

References

- Abnar, S. and Zuidema, W. Quantifying attention flow in transformers. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.385. URL <https://aclanthology.org/2020.acl-main.385/>.
- Achtibat, R., Hatefi, S. M. V., Dreyer, M., Jain, A., Wiegand, T., Lapuschkin, S., and Samek, W. AttnLRP: Attention-aware layer-wise relevance propagation for transformers. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 135–168. PMLR, 21–27 Jul 2024. URL <https://proceedings.mlr.press/v235/achtibat24a.html>.
- Ali, A., Schnake, T., Eberle, O., Montavon, G., Müller, K.-R., and Wolf, L. XAI for transformers: Better explanations through conservative propagation. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 435–451. PMLR, 2022. URL <https://proceedings.mlr.press/v162/ali22a.html>.
- Allen-Zhu, Z. and Li, Y. Physics of language models: Part 1, learning hierarchical language structures, 2024. URL <https://arxiv.org/abs/2305.13673>.
- Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., and Müller, K.-R. How to explain individual classification decisions. *The Journal of Machine Learning Research*, 11:1803–1831, 2010.
- Besta, M., Scheidl, F., Gianinazzi, L., Kwasniewski, G., Klaiman, S., Müller, J., and Hoeffler, T. Demystifying higher-order graph neural networks, 2024. URL <https://arxiv.org/abs/2406.12841>.
- Brunec, I. and Momennejad, I. Predictive representations in hippocampal and prefrontal hierarchies. *Journal of Neuroscience*, November 2021. URL <https://www.jneurosci.org/content/early/2021/11/19/JN-RM-1327-21>. JN-RM-1327-21.
- Bürger, L., Hamprecht, F. A., and Nadler, B. Truth is universal: Robust detection of lies in llms, 2024. URL <https://arxiv.org/abs/2407.12831>.
- Campbell, D., Rane, S., Giallanza, T., De Sabbata, N., Ghods, K., Joshi, A., Ku, A., Frankland, S. M., Griffiths, T. L., Cohen, J. D., et al. Understanding the limits of vision language models through the lens of the binding problem. *arXiv preprint arXiv:2411.00238*, 2024.
- Cerný, V. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985. doi: 10.1007/BF00940812.
- Chefer, H., Gur, S., and Wolf, L. Transformer interpretability beyond attention visualization. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 782–791, 2021. doi: 10.1109/CVPR46437.2021.00084.
- Choi, D., Huang, V., Meng, K., Johnson, D., Steinhardt, J., and Schwettmann, S. Scaling Automatic Neuron Description, 2024. URL <https://transluce.org/neuron-descriptions>.
- Chormai, P., Herrmann, J., Müller, K.-R., and Montavon, G. Disentangled explanations of neural network predictions by finding relevant subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

- Clark, K., Khandelwal, U., Levy, O., and Manning, C. D. What does BERT look at? an analysis of BERT’s attention. In Linzen, T., Chrupala, G., Belinkov, Y., and Hupkes, D. (eds.), *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4828. URL <https://aclanthology.org/W19-4828/>.
- Conmy, A., Mavor-Parker, A., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. In Oh, A., Nauermann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 16318–16352. Curran Associates, Inc., 2023.
- Conneau, A., Kruszewski, G., Lample, G., Barrault, L., and Baroni, M. What you can cram into a single $\&\#\&$ vector: Probing sentence embeddings for linguistic properties. In Gurevych, I. and Miyao, Y. (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2126–2136, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1198. URL <https://aclanthology.org/P18-1198/>.
- Conwell, C., Tawiah-Quashie, R., and Ullman, T. Relations, negations, and numbers: Looking for logic in generative text-to-image models. *arXiv preprint arXiv:2411.17066*, 2024.
- DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Ding, H., Xin, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Wang, J., Chen, J., Yuan, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Ye, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Zhao, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X., Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Xu, Y., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL <https://arxiv.org/abs/2501.12948>.
- Demircan, C., Saanum, T., Jagadish, A. K., Binz, M., and Schulz, E. Sparse autoencoders reveal temporal difference learning in large language models, 2024. URL <https://arxiv.org/abs/2410.01280>.
- Eberle, O., Büttner, J., Kräutli, F., Müller, K.-R., Valleriani, M., and Montavon, G. Building and interpreting deep similarity models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1149–1161, 2022. doi: 10.1109/TPAMI.2020.3020738.
- Eberle, O., Chalkidis, I., Cabello, L., and Brandl, S. Rather a nurse than a physician - contrastive explanations under investigation. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6907–6920, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.427. URL <https://aclanthology.org/2023.emnlp-main.427/>.
- Edelman, E., Tsilivis, N., Edelman, B. L., Eran Malach, and Goel, S. The evolution of statistical induction heads: In-context learning markov chains. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=qaRT6QTIqJ>.
- Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., DasSarma, N., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. <https://transformer-circuits.pub/2021/framework/index.html>.
- Elmoznino, E., Jiralspong, T., Bengio, Y., and Lajoie, G. A complexity-based theory of compositionality. *arXiv preprint arXiv:2410.14817*, 2024.

- Fang, X., Che, S., Mao, M., Zhang, H., Zhao, M., and Zhao, X. Bias of ai-generated content: an examination of news produced by large language models. *Scientific Reports*, 14(1):5224, 2024.
- Feng, J. and Steinhardt, J. How do language models bind entities in context? In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=zb3b6oK077>.
- Friedman, B. and Nissenbaum, H. Bias in computer systems. *ACM Trans. Inf. Syst.*, 14(3):330–347, July 1996. ISSN 1046-8188. doi: 10.1145/230538.230561. URL <https://doi.org/10.1145/230538.230561>.
- Fu, D., CHEN, T., Jia, R., and Sharan, V. Transformers learn higher-order optimization methods for in-context learning: A study with linear models, 2024. URL <https://openreview.net/forum?id=YKzGrt3m2g>.
- Fumagalli, F., Muschalik, M., Kolpaczki, P., Hüllermeier, E., and Hammer, B. E. SHAP-IQ: Unified approximation of any-order shapley interactions. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=IEMLNF4gK4>.
- Galke, L., Ram, Y., and Raviv, L. Deep neural networks and humans both benefit from compositional language structure. *Nature Communications*, 15:10816, 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-55158-1.
- Gandhi, K., Lee, D., Grand, G., Liu, M., Cheng, W., Sharma, A., and Goodman, N. D. Stream of search (sos): Learning to search in language. *arXiv preprint arXiv:2404.03683*, 2024.
- Geiger, A., Wu, Z., Lu, H., Rozner, J., Kreiss, E., Icard, T., Goodman, N., and Potts, C. Inducing causal structure for interpretable neural networks. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 7324–7338. PMLR, 17–23 Jul 2022. URL <https://proceedings.mlr.press/v162/geiger22a.html>.
- Gershman, S. J. Amortized inference in learning and decision making. *Current Opinion in Behavioral Sciences*, 29:80–86, 2019. doi: 10.1016/j.cobeha.2019.04.003.
- Gershman, S. J. Just looking: The innocent eye in neuroscience. *Neuron*, 109(14):2220–2223, 2021.
- Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. In Moens, M.-F., Huang, X., Specia, L., and Yih, S. W.-t. (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.446. URL <https://aclanthology.org/2021.emnlp-main.446/>.
- Giaffar, H., Rullán Buxó, C., and Aoi, M. The effective number of shared dimensions between paired datasets. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4249–4257. PMLR, 2024. URL <https://proceedings.mlr.press/v238/giaffar24a.html>.
- Gurnee, W. and Tegmark, M. Language models represent space and time, 2024. URL <https://arxiv.org/abs/2310.02207>.
- Hanna, M., Pezzelle, S., and Belinkov, Y. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. In *ICML 2024 Workshop on Mechanistic Interpretability*, 2024. URL <https://openreview.net/forum?id=grXgesr5dT>.
- Hewitt, J. and Manning, C. D. A structural probe for finding syntax in word representations. In Burstein, J., Doran, C., and Solorio, T. (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1419. URL <https://aclanthology.org/N19-1419/>.
- Huh, M., Cheung, B., Wang, T., and Isola, P. Position: The platonic representation hypothesis. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20617–20642. PMLR, 21–27 Jul 2024. URL <https://proceedings.mlr.press/v235/huh24a.html>.
- Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carney, A., et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.
- Jafari, F. R., Montavon, G., Müller, K.-R., and Eberle, O. Mambalrp: Explaining selective state space sequence models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

- Kandpal, N., Deng, H., Roberts, A., Wallace, E., and Raffel, C. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pp. 15696–15707. PMLR, 2023.
- Kauffmann, J., Esders, M., Ruff, L., Montavon, G., Samek, W., and Müller, K.-R. From clustering to cluster explanations via neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 35(2):1926–1940, 2022.
- Kauffmann, J., Dippel, J., Ruff, L., Samek, W., Müller, K.-R., and Montavon, G. The clever hans effect in unsupervised learning. *arXiv preprint arXiv:2408.08041*, 2024.
- Kaur, D., Uslu, S., Rittichier, K. J., and Duresi, A. Trustworthy artificial intelligence: A review. *ACM Comput. Surv.*, 55(2), January 2022. ISSN 0360-0300. doi: 10.1145/3491209. URL <https://doi.org/10.1145/3491209>.
- Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. Optimization by simulated annealing. *Science*, 220(4598): 671–680, 1983. doi: 10.1126/science.220.4598.671.
- Klabunde, M., Schumacher, T., Strohmaier, M., and Lemmerich, F. Similarity of neural network models: A survey of functional and representational measures, 2024. URL <https://arxiv.org/abs/2305.06329>.
- Knuth, D. E. Semantics of Context-Free Languages. *Mathematical Systems Theory*, 2(2):127–145, 1968.
- Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMLR, 2019.
- Kriegeskorte, N., Mur, M., and Bandettini, P. A. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:249, 2008.
- Lehnert, L., Sukhbaatar, S., Su, D., Zheng, Q., McVay, P., Rabbat, M., and Tian, Y. Beyond a*: Better planning with transformers via search dynamics bootstrapping. In *First Conference on Language Modeling*, 2024. URL <https://openreview.net/forum?id=SGoVIC0u0f>.
- Lepori, M., Serre, T., and Pavlick, E. Break it down: Evidence for structural compositionality in neural networks. *Advances in Neural Information Processing Systems*, 36: 42623–42660, 2023.
- Lewis, M., Nayak, N. V., Yu, P., Yu, Q., Merullo, J., Bach, S. H., and Pavlick, E. Does clip bind concepts? probing compositionality in large image models. *arXiv preprint arXiv:2212.10537*, 2022.
- Li, P., Yang, J., Islam, M. A., and Ren, S. Making ai less “thirsty”: Uncovering and addressing the secret water footprint of ai models, 2025. URL <https://arxiv.org/abs/2304.03271>.
- Li, X., Chowdhury, N., Johnson, D., Hashimoto, T., Liang, P., Schwettmann, S., and Steinhardt, J. Eliciting Language Model Behaviors with Investigator Agents, 2024a. URL <https://transluce.org/automated-elicitation>.
- Li, Y., Ildiz, M. E., Papailiopoulou, D., and Oymak, S. Transformers as algorithms: Generalization and stability in in-context learning, 2023. URL <https://arxiv.org/abs/2301.07067>.
- Li, Z., Liu, H., Zhou, D., and Ma, T. Chain of thought empowers transformers to solve inherently serial problems. *arXiv preprint arXiv:2402.12875*, 2024b.
- Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944, 2017. doi: 10.1109/CVPR.2017.106.
- Lipton, Z. C. The mythos of model interpretability, 2017. URL <https://arxiv.org/abs/1606.03490>.
- Liu, B., Ash, J. T., Goel, S., Krishnamurthy, A., and Zhang, C. Transformers learn shortcuts to automata, 2023. URL <https://arxiv.org/abs/2210.10749>.
- Lucy, L. and Bamman, D. Gender and representation bias in GPT-3 generated stories. In Akoury, N., Brahman, F., Chaturvedi, S., Clark, E., Iyyer, M., and Martin, L. J. (eds.), *Proceedings of the Third Workshop on Narrative Understanding*, pp. 48–55, Virtual, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.nuse-1.5. URL <https://aclanthology.org/2021.nuse-1.5/>.
- Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.
- Malach, E. Auto-regressive next-token predictors are universal learners. *arXiv preprint arXiv:2309.06979*, 2023.
- Manvi, R., Singh, A., and Ermon, S. Adaptive inference-time compute: LLMs can predict if they can do better, even mid-generation, 2024. URL <https://arxiv.org/abs/2410.02725>.

- Marr, D. *Vision: A computational investigation into the human representation and processing of visual information*. MIT press, 1982.
- McDougall, C. S., Conmy, A., Rushing, C., McGrath, T., and Nanda, N. Copy suppression: Comprehensively understanding a motif in language model attention heads. In Belinkov, Y., Kim, N., Jumelet, J., Mohebbi, H., Mueller, A., and Chen, H. (eds.), *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 337–363, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.blackboxnlp-1.22. URL <https://aclanthology.org/2024.blackboxnlp-1.22/>.
- Merullo, J., Eickhoff, C., and Pavlick, E. Language models implement simple Word2Vec-style vector arithmetic. In Duh, K., Gomez, H., and Bethard, S. (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5030–5047, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.281. URL <https://aclanthology.org/2024.naacl-long.281/>.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953. doi: 10.1063/1.1699114.
- Mitchell, M., Palmarini, A. B., and Moskvichev, A. Comparing humans, gpt-4, and gpt-4v on abstraction and reasoning tasks. *arXiv preprint arXiv:2311.09247*, 2023.
- Momennejad, I. Learning structures: Predictive representations, replay, and generalization. *Current Opinion in Behavioral Sciences*, 32:155–166, April 2020. URL <https://www.sciencedirect.com/science/article/pii/S2352154620300371>.
- Momennejad, I. Memory and planning in brains and machines: Multiscale predictive representations. In Nadel, L. and Aronovitz, S. (eds.), *Space, Time, and Memory*. Oxford University Press, forthcoming.
- Momennejad, I., Russek, E., Cheong, J. H., Botvinick, M. M., Daw, N., and Gershman, S. J. The successor representation in human reinforcement learning: evidence from retrospective revaluation. *Nature Human Behaviour*, 1:680–692, 2017. doi: 10.1038/s41562-017-0071. URL <https://www.nature.com/articles/s41562-017-0071>. Equal contribution.
- Momennejad, I., Hasanbeig, H., Vieira, F., Sharma, H., Ness, R. O., Jojic, N., Palangi, H., and Larson, J. Evaluating cognitive maps and planning in large language models with cogeval, 2023. URL <https://arxiv.org/abs/2309.15129>.
- Montavon, G., Samek, W., and Müller, K.-R. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018. ISSN 1051-2004. doi: <https://doi.org/10.1016/j.dsp.2017.10.011>. URL <https://www.sciencedirect.com/science/article/pii/S1051200417302385>.
- Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., and Grohe, M. Weisfeiler and leman go neural: higher-order graph neural networks. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI’19/IAAI’19/EAAI’19, pp. 4602–4609. AAAI Press, 2019. ISBN 978-1-57735-809-1. doi: 10.1609/aaai.v33i01.33014602. URL <https://doi.org/10.1609/aaai.v33i01.33014602>.
- Nanda, N., Chan, L., Lieberum, T., Smith, J., and Steinhart, J. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=9XFSbDPmdW>.
- Nisioti, E., Risi, S., Momennejad, I., Oudeyer, P.-Y., and Moulin-Frier, C. Collective innovation in groups of large language models, 2024. URL <https://arxiv.org/abs/2407.05377>.
- Olah, C. Interpretability Dreams, 2023. URL <https://transformer-circuits.pub/2023/interpretability-dreams>.
- Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., and Carter, S. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. <https://distill.pub/2020/circuits/zoom-in>.
- Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Johnston, S., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. <https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html>.
- Power, A., Burda, Y., Edwards, H., Babuschkin, I., and Misra, V. Grokking: Generalization beyond overfit-

- ting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.
- Qiu, L., Shaw, P., Pasupat, P., Shi, T., Herzig, J., Pitler, E., Sha, F., and Toutanova, K. Evaluating the impact of model scale for compositional generalization in semantic parsing. In *Proc. of Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9157–9179. ACL, December 2022. doi: 10.18653/v1/2022.emnlp-main.624.
- Radev, S. T., Voss, A., Wieschen, E. M., and Bürkner, P.-C. Amortized bayesian inference for models of cognition, 2020. URL <https://arxiv.org/abs/2005.03899>.
- Ren, J., Guo, Q., Yan, H., Liu, D., Zhang, Q., Qiu, X., and Lin, D. Identifying semantic induction heads to understand in-context learning. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6916–6932, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.412. URL <https://aclanthology.org/2024.findings-acl.412/>.
- Russek, E., Momennejad, I., Botvinick, M. M., Gershman, S. J., and Daw, N. Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLOS Computational Biology*, 13(6): e1005474, 2017. doi: 10.1371/journal.pcbi.1005474. URL <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005474>.
- Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., and Müller, K.-R. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278, 2021. doi: 10.1109/JPROC.2021.3060483.
- Sanford, C., Hsu, D., and Telgarsky, M. Representational strengths and limitations of transformers, 2023. URL <https://arxiv.org/abs/2306.02896>.
- Sanford, C., Fatemi, B., Hall, E., Tsitsulin, A., Kazemi, M., Halcrow, J., Perozzi, B., and Mirrokni, V. Understanding transformer reasoning capabilities via graph algorithms. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=AfzbDw6DSp>.
- Satorras, V. G., Hoogeboom, E., and Welling, M. E(n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.
- Schnake, T., Eberle, O., Lederer, J., Nakajima, S., Schütt, K. T., Müller, K.-R., and Montavon, G. Higher-order explanations of graph neural networks via relevant walks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7581–7596, 2022. doi: 10.1109/TPAMI.2021.3115452.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms, 2017. URL <https://arxiv.org/abs/1707.06347>.
- Shah, D. S., Schwartz, H. A., and Hovy, D. Predictive biases in natural language processing models: A conceptual framework and overview. In Jurafsky, D., Chai, J., Schluter, N., and Tetraault, J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5248–5264, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.468. URL <https://aclanthology.org/2020.acl-main.468/>.
- Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y. K., Wu, Y., and Guo, D. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL <https://arxiv.org/abs/2402.03300>.
- Sharma, A., Czégel, D., Lachmann, M., Kempes, C. P., Walker, S. I., and Cronin, L. Assembly theory explains and quantifies selection and evolution. *Nature*, 622(7982): 321–328, 2023.
- Shneidman, J. and Parkes, D. C. Specification faithfulness in networks with rational nodes. In *Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Distributed Computing*, PODC ’04, pp. 88–97, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138024. doi: 10.1145/1011767.1011781. URL <https://doi.org/10.1145/1011767.1011781>.
- Silver, D., Singh, S., Precup, D., and Sutton, R. S. Reward is enough. *Artificial Intelligence*, 299:103535, 2021. ISSN 0004-3702. doi: <https://doi.org/10.1016/j.artint.2021.103535>. URL <https://www.sciencedirect.com/science/article/pii/S0004370221000862>.
- Snell, C., Lee, J., Xu, K., and Kumar, A. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.
- Stechly, K., Valmeekam, K., and Kambhampati, S. Chain of thoughtlessness: An analysis of cot in planning. *arXiv preprint arXiv:2405.04776*, 2024.

- Strobl, L., Merrill, W., Weiss, G., Chiang, D., and Angluin, D. What formal languages can transformers express? a survey. *Transactions of the Association for Computational Linguistics*, 12:543–561, 2024. ISSN 2307-387X. doi: 10.1162/tacl_a_00663. URL http://dx.doi.org/10.1162/tacl_a_00663.
- Strubell, E., Ganesh, A., and McCallum, A. Energy and policy considerations for deep learning in nlp, 2019. URL <https://arxiv.org/abs/1906.02243>.
- Sucholutsky, I., Muttenthaler, L., Weller, A., Peng, A., Bobu, A., Kim, B., Love, B. C., Cueva, C. J., Grant, E., Groen, I., Achterberg, J., Tenenbaum, J. B., Collins, K. M., Hermann, K. L., Oktar, K., Greff, K., Hebart, M. N., Cloos, N., Kriegeskorte, N., Jacoby, N., Zhang, Q., Marjeh, R., Geirhos, R., Chen, S., Kornblith, S., Rane, S., Konkle, T., O’Connell, T. P., Unterthiner, T., Lampinen, A. K., Müller, K.-R., Toneva, M., and Griffiths, T. L. Getting aligned on representational alignment, 2024. URL <https://arxiv.org/abs/2310.13018>.
- Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.
- Sutton, R. The bitter lesson, 2019. URL <http://www.incompleteideas.net/IncIdeas/BitterLesson.html>.
- Swartout, W. R. and Moore, J. D. Explanation in second generation expert systems. In David, J.-M., Krivine, J.-P., and Simmons, R. (eds.), *Second Generation Expert Systems*, pp. 543–585, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg. ISBN 978-3-642-77927-5.
- Syed, A., Rager, C., and Conmy, A. Attribution patching outperforms automated circuit discovery. In Belinkov, Y., Kim, N., Jumelet, J., Mohebbi, H., Mueller, A., and Chen, H. (eds.), *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 407–416, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.blackboxnlp-1.25. URL <https://aclanthology.org/2024.blackboxnlp-1.25/>.
- Talon, D., Lippe, P., James, S., Bue, A. D., and Magliacane, S. Towards the reusability and compositionality of causal representations. In Locatello, F. and Didelez, V. (eds.), *Proceedings of the Third Conference on Causal Learning and Reasoning*, volume 236 of *Proceedings of Machine Learning Research*, pp. 296–324. PMLR, 01–03 Apr 2024. URL <https://proceedings.mlr.press/v236/talon24a.html>.
- Tarzanagh, D. A., Li, Y., Thrampoulidis, C., and Oymak, S. Transformers as support vector machines. In *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*, 2023. URL <https://openreview.net/forum?id=gLwzzmh79K>.
- Templeton, A., Conerly, T., Marcus, J., Lindsey, J., Bricken, T., Chen, B., Pearce, A., Citro, C., Ameisen, E., Jones, A., Cunningham, H., Turner, N. L., McDougall, C., MacDiarmid, M., Freeman, C. D., Summers, T. R., Rees, E., Batson, J., Jermyn, A., Carter, S., Olah, C., and Henighan, T. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL <https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html>.
- Tigges, C., Hanna, M., Yu, Q., and Biderman, S. Llm circuit analyses are consistent across training and scale, 2024. URL <https://arxiv.org/abs/2407.10827>.
- Todd, E., Li, M., Sharma, A. S., Mueller, A., Wallace, B. C., and Bau, D. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=AwyxtyMwaG>.
- Tsai, Y.-H. H., Bai, S., Yamada, M., Morency, L.-P., and Salakhutdinov, R. Transformer dissection: An unified understanding for transformer’s attention via the lens of kernel. In Inui, K., Jiang, J., Ng, V., and Wan, X. (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4344–4353, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1443. URL <https://aclanthology.org/D19-1443/>.
- Turing, A. M. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 2(42):230–265, 1936. URL <http://www.cs.helsinki.fi/u/gionis/cc05/OnComputableNumbers.pdf>.
- Turpin, M., Michael, J., Perez, E., and Bowman, S. Language models don’t always say what they think: unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36, 2024.
- Vasileiou, A. and Eberle, O. Explaining text similarity in transformer models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7859–7873, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.

435. URL <https://aclanthology.org/2024.naacl-long.435/>.
- Vaswani, A. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- Veličković, P. and Blundell, C. Neural algorithmic reasoning. *Patterns*, 2(7), 2021.
- Vig, J. and Belinkov, Y. Analyzing the structure of attention in a transformer language model, 2019. URL <https://arxiv.org/abs/1906.04284>.
- Vilas, M. G., Adolphi, F., Poeppel, D., and Roig, G. Position: An inner interpretability framework for AI inspired by lessons from cognitive neuroscience. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 49506–49522. PMLR, 21–27 Jul 2024. URL <https://proceedings.mlr.press/v235/vilas24a.html>.
- Villalobos, P., Ho, A., Sevilla, J., Besiroglu, T., Heim, L., and Hobbhahn, M. Position: will we run out of data? limits of llm scaling based on human-generated data. In *Proceedings of the 41st International Conference on Machine Learning*, ICML’24. JMLR.org, 2024.
- von Oswald, J., Schlegel, M., Meulemans, A., Kobayashi, S., Niklasson, E., Zucchet, N., Scherrer, N., Miller, N., Sandler, M., y Arcas, B. A., Vladymyrov, M., Pascanu, R., and Sacramento, J. Uncovering mesa-optimization algorithms in transformers, 2024. URL <https://arxiv.org/abs/2309.05858>.
- Wang, K. R., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=NpsVSN6o4ul>.
- Webb, T., Mondal, S. S., and Momennejad, I. Improving planning with large language models: A modular agentic architecture, 2024. URL <https://arxiv.org/abs/2310.00194>.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Weiss, G., Goldberg, Y., and Yahav, E. Thinking like transformers. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 11080–11090. PMLR, 18–24 Jul 2021. URL <https://proceedings.mlr.press/v139/weiss21a.html>.
- Wiedemer, T., Mayilvahanan, P., Bethge, M., and Brendel, W. Compositional generalization from first principles. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=LqOQluJmSx>.
- Wiegrefe, S. and Pinter, Y. Attention is not not explanation. In Inui, K., Jiang, J., Ng, V., and Wan, X. (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 11–20, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1002. URL <https://aclanthology.org/D19-1002/>.
- Williams, A. H., Kunz, E., Kornblith, S., and Linderman, S. Generalized shape metrics on neural representations. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 4738–4750. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/252a3dbaeb32e7690242ad3b556e626b-Paper.pdf.
- Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D., and Wang, C. Autogen: Enabling next-gen llm applications via multi-agent conversation, 2023a. URL <https://arxiv.org/abs/2308.08155>.
- Wu, Z., Geiger, A., Icard, T., Potts, C., and Goodman, N. Interpretability at scale: Identifying causal mechanisms in alpaca. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 78205–78226. Curran Associates, Inc., 2023b.
- Yang, W., Sun, C., and Buzsaki, G. INTERPRETABILITY OF LLM DECEPTION: UNIVERSAL MOTIF. In *Neurips Safe Generative AI Workshop 2024*, 2024. URL <https://openreview.net/forum?id=DRWCDFsb2e>.
- Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36, 2024.
- Ye, T., Xu, Z., Li, Y., and Allen-Zhu, Z. Physics of Language Models: Part 2.1, Grade-School Math and

- the Hidden Reasoning Process. In *Proceedings of the 13th International Conference on Learning Representations*, ICLR '25, April 2025. Full version available at <http://arxiv.org/abs/2407.20311>.
- Yousefi, S., Betthausen, L., Hasanbeig, H., Millière, R., and Momennejad, I. Decoding in-context learning: Neuroscience-inspired analysis of representations in large language models, 2024. URL <https://arxiv.org/abs/2310.00313>.
- Yu, Q., Merullo, J., and Pavlick, E. Characterizing mechanisms for factual recall in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9924–9959, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.615. URL <https://aclanthology.org/2023.emnlp-main.615/>.
- Zekri, O., Odonnat, A., Benechehab, A., Bleistein, L., Boullé, N., and Redko, I. Large language models as markov chains, 2024. URL <https://arxiv.org/abs/2410.02724>.
- Zhong, Z., Liu, Z., Tegmark, M., and Andreas, J. The clock and the pizza: Two stories in mechanistic explanation of neural networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=S5wmbQc1We>.
- Zhou, B., Bau, D., Oliva, A., and Torralba, A. Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2131–2145, 2018.
- Zhou, H., Bradley, A., Littwin, E., Razin, N., Saremi, O., Susskind, J. M., Bengio, S., and Nakkiran, P. What algorithms can transformers learn? a study in length generalization. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=AssIuHnmHX>.
- Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., Goel, S., Li, N., Byun, M. J., Wang, Z., Mallen, A., Basart, S., Koyejo, S., Song, D., Fredrikson, M., Kolter, J. Z., and Hendrycks, D. Representation engineering: A top-down approach to ai transparency, 2023. URL <https://arxiv.org/abs/2310.01405>.

A. Appendix

A.1. Statistical Testing: Paired-sample t -tests for attention from the final token to correct vs. incorrect pathways.

Table 1: Reports statistics for the linear mixed-effects model, testing attention directed from the final token to the correct versus incorrect pathway across all layers. Mixed-effects modeling was conducted using the lmerTest package in R.

Table 2: Reports specific layers with significantly greater attention to the correct pathway, while Tab. 3 reports layers with significantly greater attention to the incorrect pathway. Note that conducting multiple t -tests (one for each layer) increases the risk of Type I errors (false positives). With k layers, we have k chances to incorrectly reject the null hypothesis. To mitigate the possibility of Type I errors, we used Bonferroni to control the family-wise error rate.

Table 1. Greater attention to the correct pathway across layers.

b	SE	t -statistic	df	p -value
0.33	0.07	4.51	2015	$p = 7.0 \times 10^{-6}$

Table 2. Greater attention to the correct pathway (individual layers).

Layer	t -Statistic	df	p -Value
6	5.68	31	$p = 1.5 \times 10^{-6}$
7	12.18	31	$p = 1.17 \times 10^{-13}$
10	9.47	31	$p = 5.86 \times 10^{-11}$
11	8.97	31	$p = 1.99 \times 10^{-10}$
12	4.32	31	$p = 7.49 \times 10^{-5}$
13	7.13	31	$p = 2.60 \times 10^{-8}$
14	3.84	31	$p = 2.87 \times 10^{-4}$
16	4.36	31	$p = 6.65 \times 10^{-5}$
17	4.60	31	$p = 3.40 \times 10^{-5}$
20	3.71	31	$p = 4.1 \times 10^{-4}$
21	4.73	31	$p = 2.36 \times 10^{-5}$
26	3.38	31	$p = 9.9 \times 10^{-4}$
29	5.96	31	$p = 6.97 \times 10^{-7}$
30	4.00	31	$p = 1.83 \times 10^{-4}$

Table 3. Greater attention to the incorrect pathway (individual layers).

Layer	t -statistic	df	p -value
0	-5.34	31	$p = 4.05 \times 10^{-6}$
2	-4.58	31	$p = 3.52 \times 10^{-5}$
23	-3.23	31	$p = 1.47 \times 10^{-3}$

A.2. Attention from the Goal Location to all Nodes for the Tree Graph (n=7)

Prompts

(1) From the hotel lobby, there are various rooms denoted by letter names. The lobby connects to O, which connects to W and Q. The lobby also connects to G, which connects to V and M. How can someone get to **W/Q/V/M** from the lobby?

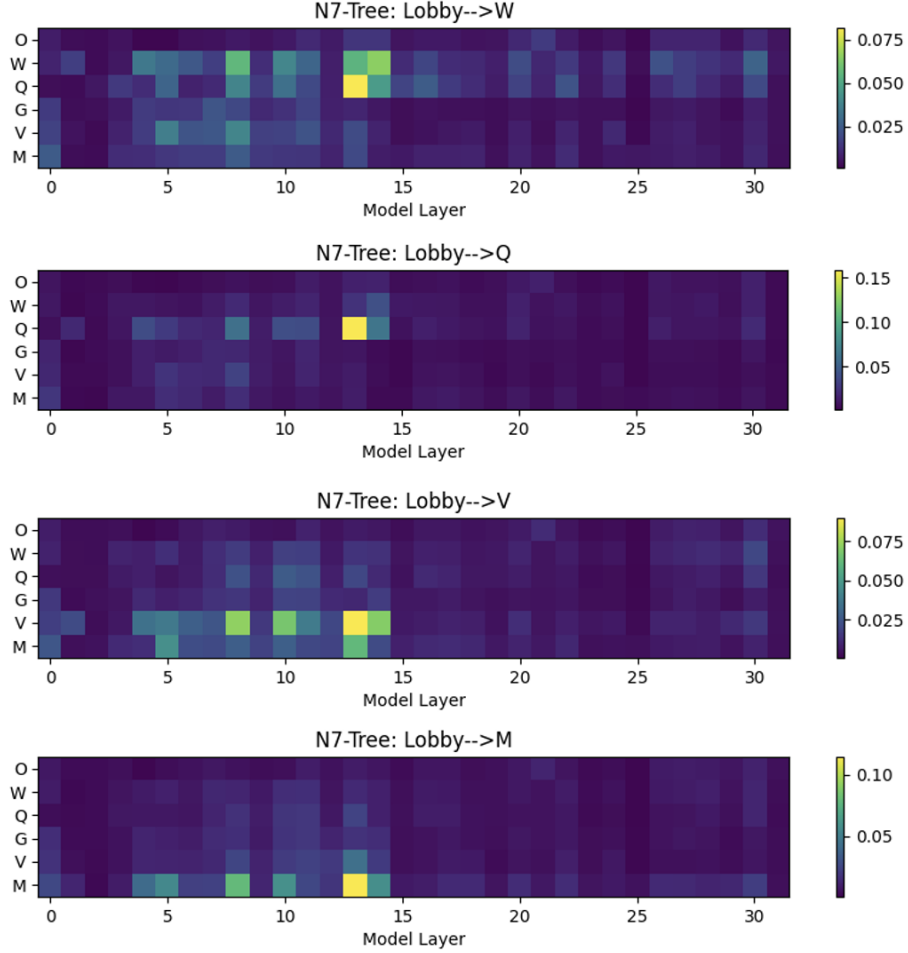


Figure 4. Attention heatmaps from the goal token to all graph nodes in the tree graph, when each final node is specified as the goal location.

A.3. Analyzing Representations

Prompts

(1) Given a description of a set of rooms, determine if someone can get to W from the lobby? Only answer "Yes" or "No". The description is: From the hotel lobby, there are various rooms denoted by letter names. The lobby connects to O, which connects to W and Q. The lobby also connects to G, which connects to V and M. Answer:

(2) Given a description of a set of rooms, determine if someone can get to Q from the lobby? Only answer "Yes" or "No". The description is: From the hotel lobby, there are various rooms denoted by letter names. The lobby connects to O, which connects to W and Q. The lobby also connects to G, which connects to V and M. Answer:

(3) Given a description of a set of rooms, determine if someone can get to V from the lobby? Only answer "Yes" or "No". The description is: From the hotel lobby, there are various rooms denoted by letter names. The lobby connects to O, which connects to W and Q. The lobby also connects to G, which connects to V and M. Answer:

(4) Given a description of a set of rooms, determine if someone can get to M from the lobby? Only answer "Yes" or "No". The description is: From the hotel lobby, there are various rooms denoted by letter names. The lobby connects to O, which connects to W and Q. The lobby also connects to G, which connects to V and M. Answer:

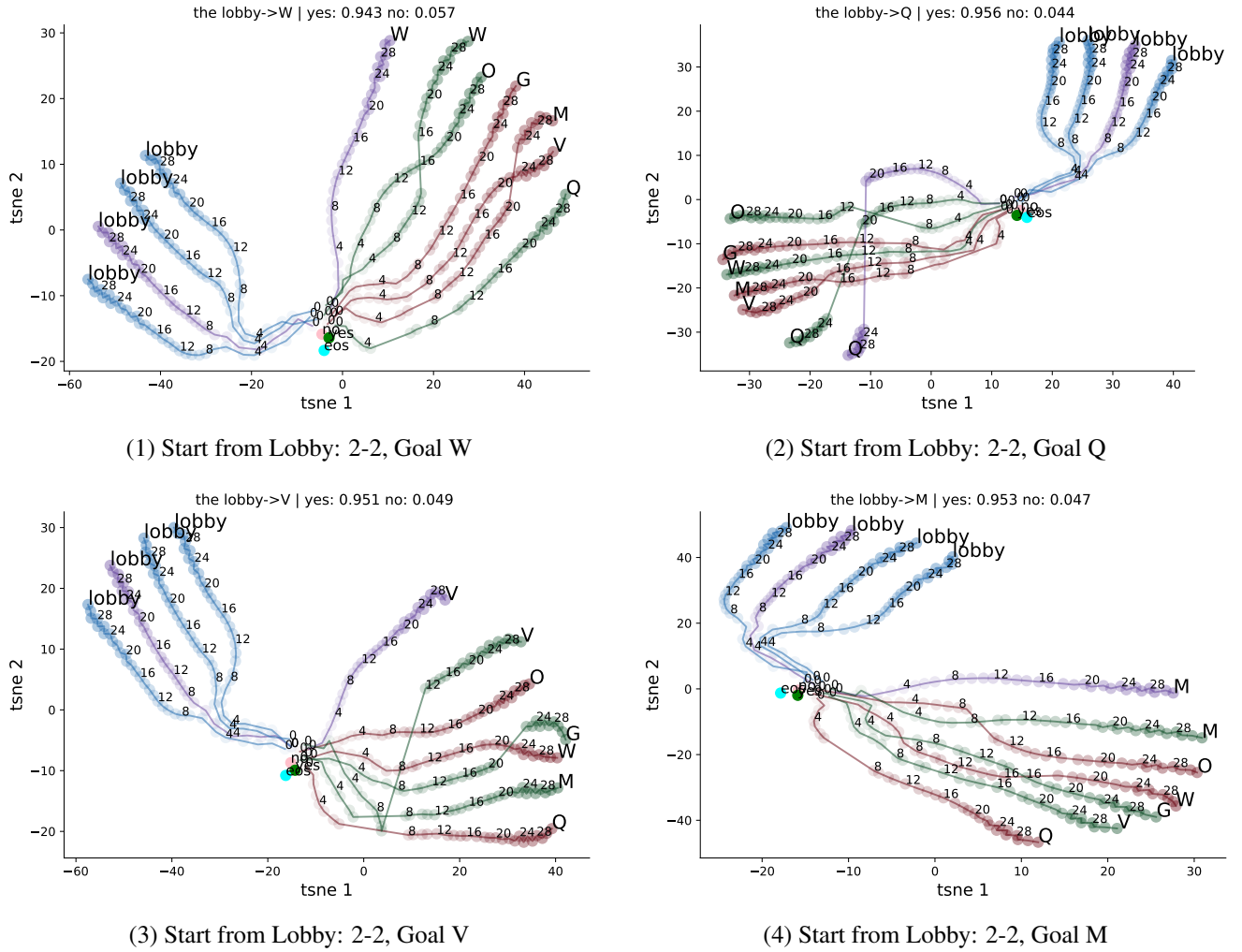


Figure 5. T-SNE projections of activations in a graph search setting. Plots correspond to prompts (1)–(4), corresponding to a changing goal node in a tree graph of $n=7$ nodes (see Figure 2). The probabilities of generating the correct 'yes' token as compared to the 'no' token using the final token representation is given for each setting.

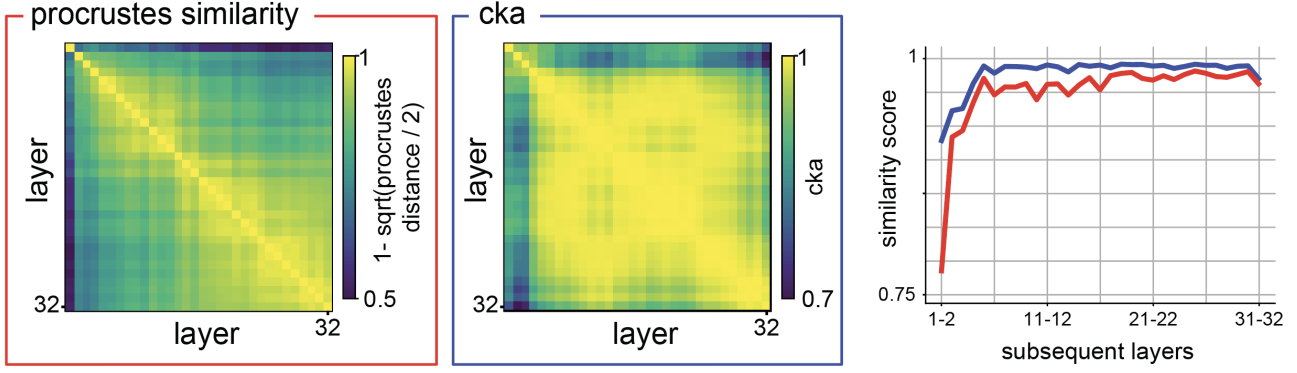


Figure 6. *Left*: We reasoned that discrete steps in a BFS or DFS rollout might be identified as substantial changes in representational geometry between subsequent layers. To explore this possibility, we computed representational similarity matrices using two similarity measures: the procrustes similarity (Williams et al., 2021) and the centered kernel alignment (Kornblith et al., 2019). *Right*: similarity scores across subsequent layers.

A.4. Results on Llama3.1-70B-Instruct

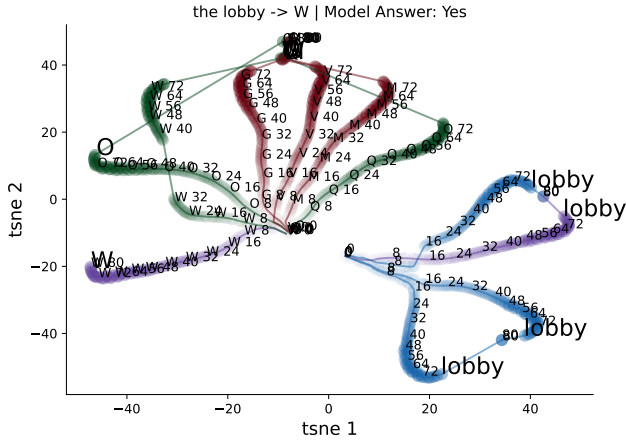


Figure 7. Representational analysis using Llama3.1-70B. Our t-SNE results show a clear separation of room representations across layers, in particular, a separation of non-goal nodes (green & red) vs. the goal node ('W' in purple). See also Figure 3 in the main paper for details.

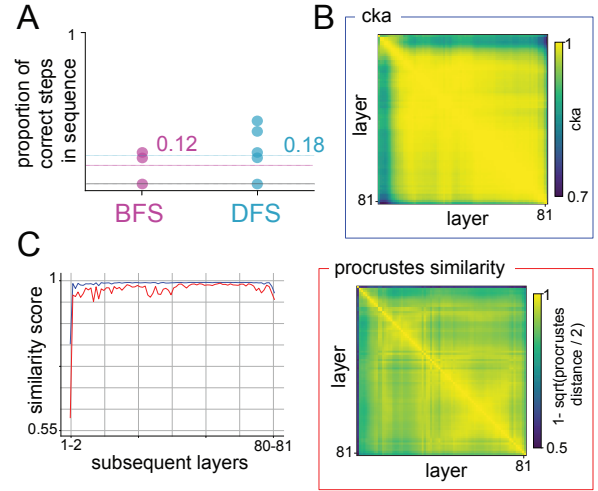


Figure 8. Evaluation of breadth-first search (BFS) and depth-first search (DFS) hypotheses in Llama3.1-70B. (A) Proportion of correct steps identified in the layer by layer hidden activations. Each data point represents a single rollout of the BFS or DFS algorithm. (B) Representational similarities between layers computed using two similarity measures. (C) Representational similarity between subsequent layers (off diagonal).

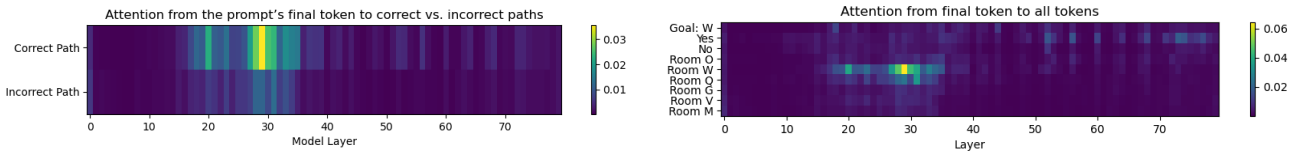


Figure 9. Attention heatmaps for Llama-3.1-70B. *Left*: Average attention per layer directed from the final token to the correct vs. incorrect pathways when node W is specified as the goal location. *Right*: Average attention from final token to different graph node and response tokens.