UNDERSTANDING CONTEXTUAL RECALL IN TRANSFORMERS: HOW FINETUNING ENABLES IN-CONTEXT REASONING OVER PRETRAINING KNOWLEDGE

Anonymous authors
Paper under double-blind review

ABSTRACT

Transformer-based language models excel at in-context learning (ICL), where they can adapt to new tasks based on contextual examples, without parameter updates. In a specific form of ICL, which we refer to as *contextual recall*, models pretrained on open-ended text leverage pairwise examples to recall specific facts in novel prompt formats. We investigate whether contextual recall emerges from pretraining alone, what finetuning is required, and what mechanisms drive the necessary representations. We introduce a controlled synthetic framework where pretraining sequences consist of subject-grammar-attribute tuples, with attribute types tied to grammar statistics. We demonstrate that while such pretraining successfully yields factual knowledge, it is insufficient for contextual recall: models fail to implicitly infer attribute types when the grammar statistics are removed in ICL prompts. However, we show that finetuning on tasks that require implicit inference, but are distinct from the ICL evaluation, using a subset of subjects, triggers the emergence of contextual recall. This transition is accompanied by the formation of low-dimensional latent encodings of the shared attribute type. For mechanistic insight, we derive a construction for an attention-only model to succeed on factual and contextual recall, corroborated by empirical validation, and analysis of the training dynamics for the transition from factual to contextual recall.

1 INTRODUCTION

Transformer-based large language models (LLMs) exhibit remarkable abilities to extrapolate far beyond tasks seen during training. A notable instance of this extrapolation is in-context learning (ICL) (Brown et al., 2020), where models can adapt to new tasks based on contextual examples, without parameter updates. In this paper, we investigate a specific form of ICL, which we refer to as *contextual recall*. Here, a model trained on open-ended text acquires factual knowledge and is later able to recall specific facts when prompted with example pairs in an unseen format. To illustrate, pretraining data might include descriptions of various landmarks, from which the model learns multiple attributes for each—such as the country where it is located, the year it was built, or its architectural style. At test time, the model receives a prompt of the form `[Niagara Falls, Canada. Colosseum, Italy. Parthenon, ]` and must generate `[Greece]`. The prompt contains no explicit indication that the relevant attribute is "country" rather than, say, "year built"; the model must infer the attribute type from the in-context examples and recall the corresponding fact for the query subject.

The ability to succeed at contextual recall therefore requires both the acquisition of factual knowledge and adaptability to novel prompt formats, since the model may never have seen these facts presented as implicit subject-attribute pairs during pretraining. We investigate the origins of this capability:

Does the ability to do contextual recall emerge naturally from pretraining, or does it necessitate specific finetuning? What mechanisms within the Transformer architecture enable its emergence?

Prior work has leveraged controlled, synthetic settings to understand both ICL and factual recall in transformers trained from scratch. For ICL, various studies have considered well-defined tasks, such as linear regression (Garg et al., 2022; Akyürek et al., 2023; von Oswald et al., 2022; Zhang et al., 2023; Fu et al., 2024), discrete functions (Bhattamishra et al., 2024), and Markov processes (Bietti et al., 2023; Edelman et al., 2024b; Rajaraman et al., 2024; Park et al., 2025; Deora et al., 2025) that involve function induction, but do not require recalling learned factual information. Importantly, and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

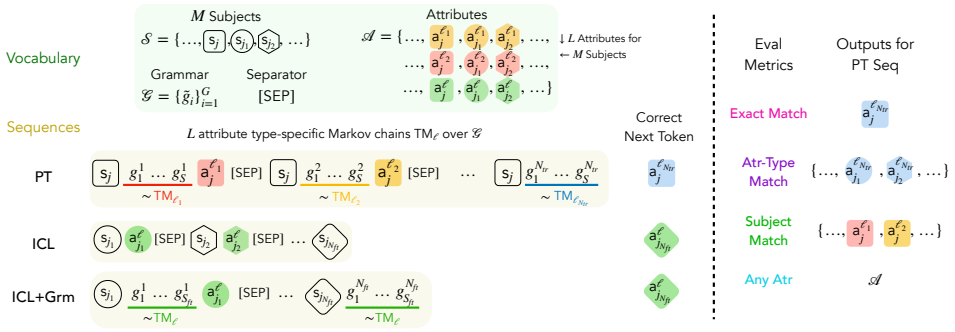

Figure 1: Illustration of the data generation process and the evaluation metrics. The vocabulary contains M subjects with L attributes each (attribute tokens may be shared across subjects), G grammar tokens and a separator token. We sample L Markov chains, one for each attribute type. We consider 3 types of sequences (see Sec. 2 for details): `PT` sequences contain subject-grammar-attribute tuples, with attribute type information encoded in the grammar sequence statistics, for one subject, `ICL` sequences contain pairwise subject-attribute examples for a shared attribute type, but no grammar, and `ICL+Grm` sequences are analogous to `ICL` sequences, but contain subject-grammar-attribute tuples. We evaluate whether the model's prediction (on `PT` seqs.) matches the ground truth attribute (exact match), any attribute of the same type as the last subsequence (attribute-type match), any attribute of the same subject as the sequence (subject match), or any attribute token (attribute match).

in contrast to our contextual recall task, the train and test data share the sequence formats in these settings. On the other hand, in studies focused on question-answering and recall tasks (Allen-Zhu & Li, 2023; Behnia et al., 2025; Nichani et al., 2024b; Zucchet et al., 2025), the context contains explicit information (e.g., specific words, or linguistic patterns) required to recall specific learned facts, and does not require the model to perform any implicit inference (e.g., what fact is relevant based on the in-context examples) that is characteristic of ICL. (See App. A for a detailed discussion on related work.)

Contributions. We summarize our contributions below.

First, in Section 2, we introduce a controlled synthetic framework to study the emergence of contextual recall in transformers. The pretraining (`PT`) sequences, used to instill factual knowledge in the model, contain multiple attributes of a single subject, interspersed with grammar tokens that encode information about each attribute type. The `ICL` sequences, used to evaluate contextual recall ability, consist of subject-attribute pairs sharing a common attribute type across different subjects, without any grammar (see Fig. 1 for an illustration of the data generation process).

In Section 3.1, we show that transformers trained with `PT` sequences succeed on factual recall, but fail to generalize to `ICL` sequences. However, we find that finetuning the model on sequences that are distinct from `ICL` sequences, but require implicit inference, using a subset of subjects, enables out-of-distribution generalization on `ICL` sequences on the held-out subjects (Section 3.2). We also probe the effect of some dataset parameters on the model's performance on `PT` and `ICL` sequences.

In Section 3.3, we analyze model representations and find that finetuning induces the formation of low-dimensional encodings of the shared attribute type (task vectors) based on the in-context examples, and they become more disentangled as the number of in-context examples increases.

Finally, in Section 4, for mechanistic insight, we consider a simpler synthetic task and present constructions for an attention-only model that succeeds on factual recall after pretraining and contextual recall after finetuning, and corroborate them with an analysis of the training dynamics for finetuning as well as empirical validation.

2 Data Generation Process

To study contextual recall in transformers, we define three different types of sequences for pretraining, finetuning, and/or evaluation. The first, `PT`, is designed to instill factual knowledge, while the other two, `ICL` and `ICL+Grm`, are used to evaluate or facilitate how the model leverages such knowledge for in-context reasoning tasks. See Fig. 1 for an illustration.

We first introduce some useful notation. Let $\mathcal{S} = \{s_j\}_{j=1}^{M}$ denote the set of M subjects (e.g., landmarks such as `Parthenon` ($j=1$), `Colosseum` ($j=2$)). Let the set of unique attributes be denoted as $\mathcal{U} = \cup_{\ell=1}^{L} \mathcal{U}_\ell$, where $\mathcal{U}_\ell = \{u_i^\ell\}_{i=1}^{M_\ell}$, and ℓ indexes the attribute type (e.g., $\ell=1$ for "country", $\ell=2$ for "year built"), and M_ℓ denotes the number of unique values for index ℓ (e.g., {`Greece`, `Italy`, ...} for "country"). Next, let $\mathcal{A} = \cup_{j=1}^{M} \{a_j^1, \ldots, a_j^L\}$ denote the set of attributes assigned to the subjects, where a_j^ℓ is the type-ℓ attribute of subject s_j (e.g., $a_2^1 =$ `Italy`). Each subject has one attribute per type, giving L attributes per subject. Let $\mathcal{G} = \{\tilde{g}_1, \ldots, \tilde{g}_G\}$ denote a set of G grammar tokens, and [sep] denote the separator token. The full vocabulary is $\mathcal{V} = \mathcal{S} \cup \mathcal{U} \cup \mathcal{G} \cup \{[\text{sep}]\}$, with size $V = M + \sum_{\ell=1}^{L} M_\ell + G + 1$. Let $\text{Unif}(\cdot)$ denote the uniform distribution, and $||$ denote concatenation.

With this notation established, we now describe the data generation process. The key building block is a *subject-grammar-attribute subsequence*. For subject s_j, attribute type ℓ, and grammar sequence length S, it is denoted as $X_j^\ell = [s_j, g_1, \ldots, g_S, a_j^\ell]$ (e.g., [`Parthenon`, *one*, *of*, *the*, *wonders*, ..., *is in*, `Greece`]). The grammar sequence $g_{1:S}$ is generated using a first-order Markov chain specific to attribute type ℓ. That is, $p(g_t = \tilde{g} | g_{t-1}, \ldots, g_1) = p(g_t = \tilde{g} | g_{t-1})$ for all $\tilde{g} \in \mathcal{G}$, with the first token drawn uniformly at random. For each attribute type $\ell \in [L]$, we sample a row-stochastic transition matrix $\text{TM}_\ell \in \mathbb{R}^{G \times G}$, where each row is drawn independently from a Dirichlet prior. Since each attribute type has its own Markov chain, the bigram statistics of the grammar sequence $g_{1:S}$ implicitly encode information about the attribute type ℓ.

We now define the three types of sequences used for pretraining, finetuning, and/or evaluation.

`PT` Sequences (Pretraining). To instill factual knowledge in the model, we use `PT` sequences which contain information about a specific subject and its associated attributes (analogous to a short encyclopedia entry for a given landmark). Let N_{tr} denote the number of subsequences (subject-grammar-attribute tuples) in each sequence. To generate a `PT` sequence, we first sample a subject $j \sim \text{Unif}([M])$, then draw N_{tr} attribute types $\ell_{1:N_{\text{tr}}} \sim \text{Unif}([L])^{N_{\text{tr}}}$, and sample each grammar subsequence $g_{1:S}^i$ from the corresponding Markov chain TM_{ℓ_i}. The resulting sequence is $\tilde{X} = X_j^{\ell_1} || [\text{sep}] || X_j^{\ell_2} \ldots X_j^{\ell_{N_{\text{tr}}}}$, with sequence length $T = (S+3)N_{\text{tr}} - 1$. To model generic text that does not convey factual information, we also include grammar-only subsequences: with probability p_G, each subject-grammar-attribute subsequence is replaced by a grammar-only subsequence $g_{1:S+2}$, generated from a separate fixed Markov chain (with transition matrix sampled from a Dirichlet prior).

`ICL` Sequences (Finetuning and Evaluation). To test for contextual recall, we use `ICL` sequences that contain subject-attribute pairs with a shared attribute type across different subjects—mirroring the evaluation format from the introduction (e.g., [`Colosseum`, `Italy`, [sep], ..., `Parthenon`,]). Let N denote the number of in-context demonstrations. We sample an attribute type $\ell \sim \text{Unif}([L])$, and $N+1$ subjects $j_{1:N+1} \sim \text{Unif}([M])^{N+1}$. The resulting sequence is $X = [s_{j_1}, a_{j_1}^\ell, [\text{sep}], \ldots, s_{j_{N+1}}, a_{j_{N+1}}^\ell]$. At evaluation, the model observes N demonstrations and must predict the attribute $a_{j_{N+1}}^\ell$ for the final subject $s_{j_{N+1}}$, without any grammar cues indicating the attribute type.

`ICL+Grm` Sequences (Finetuning). We also use `ICL+Grm` sequences for finetuning. As the name suggests, these sequences are similar to `ICL` sequences (subject-attribute pairs sharing a common attribute type across different subjects), but they also contain grammar tokens between the subject and attribute. (As we will see in Section 3.2, these sequences help instill the implicit inference ability required to succeed at contextual recall on `ICL` sequences.) Let S_{ft} denote the grammar sequence length used in each subsequence, and N_{ft} denote the number of subsequences. We sample attribute type $\ell \sim \text{Unif}([L])$ and subjects $j_{1:N_{\text{ft}}+1} \sim \text{Unif}([M])^{N_{\text{ft}}+1}$, then generate grammar subsequences $g_{1:S_{\text{ft}}}^i$ from the corresponding Markov chain TM_ℓ. The resulting `ICL+Grm` sequence is $X = X_{j_1}^\ell || [\text{sep}] || X_{j_2}^\ell \ldots X_{j_{N_{\text{ft}}+1}}^\ell$. `ICL` sequences are a special case of `ICL+Grm` with $S_{\text{ft}} = 0$.

3 Experimental Results

3.1 Pretraining on `PT` Sequences

Setup. We train a two-layer, single-head, decoder-only transformer on `PT` sequences to minimize the standard next-token prediction objective. We use an online training setup: at each iteration, we generate a fresh batch of `PT` sequences using the process described in Sec. 2. Unless stated otherwise, we fix $M = 256$ subjects, $L = 8$ attribute types, and $S = 80$ grammar length (see App. B for details).

We evaluate the model on two distinct sets: i) PT sequences, to evaluate factual recall, and ii) ICL sequences, to evaluate contextual recall capabilities. For both, we measure the accuracy of predicting the final token given the preceding context. We report four metrics: *Exact Match*, *Attribute-Type Match*, *Subject Match*, and *Attribute Match* (see Fig. 1 for an illustration).

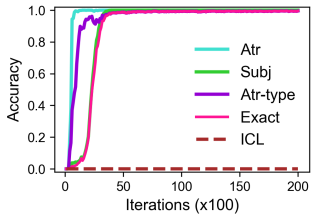

Figure 2: Transformer trained on PT sequences performs well on PT sequences, but does not generalize to ICL sequences.

Fig. 2 illustrates the performance as pretraining progresses. On the PT sequences (solid lines), we observe a stage-wise learning process: the model first learns to predict the attribute type, and then the exact attribute token. Crucially, however, we find that high performance on PT sequences does not transfer to ICL sequences, with the model achieving near-zero accuracy on ICL sequences. This shows that despite learning the factual associations, the model relies on *explicit* grammar statistics for retrieval and cannot *implicitly* infer the attribute type from in-context examples alone. We summarize this in our first main finding.

Finding 1: Pretraining on PT sequences does *not* suffice for good ICL performance.

3.2 Finetuning Experiments

Since pretraining on PT sequences alone does not suffice for good ICL performance, we investigate whether finetuning can bridge this gap. Specifically, we ask: can a model originally trained to rely on explicit grammar-based cues be adapted to perform implicit inference from in-context examples? To answer this, we finetune the pretrained model using the standard next-token prediction objective on a new data distribution (detailed below). Crucially, to test for generalization, we finetune on only a subset of subjects, reserving the remaining subjects as a held-out set. We then evaluate the model on ICL sequences (as defined in Section 2) where the query subject belongs to this held-out set. We set the number of demonstrations $N_{\text{ft}} = 16$, and use 50% of the total subjects for finetuning.

FT on ICL using a subset of subjects. We first consider the most direct approach: finetuning the pretrained model directly on ICL sequences. This serves as an upper bound, since the finetuning and evaluation formats are identical, *i.e.*, there is no distribution shift in prompt format. Fig. 3 shows performance on ICL sequences with seen and held-out query subjects. As finetuning progresses, the model successfully learns to perform contextual recall, generalizing to held-out subjects. This is our second main finding.

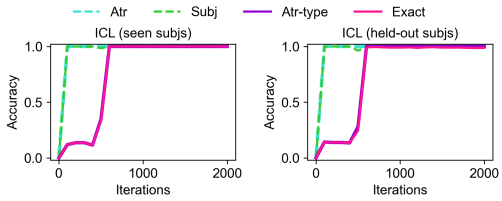

Figure 3: Finetuning the model pretrained on PT sequences, using ICL sequences with a subset of subjects, leads to good performance on ICL sequences with held-out query subjects.

Finding 2: Finetuning on ICL sequences with a subset of subjects leads to generalization on ICL sequences with held-out subjects.

Mechanism of Transfer. Why does finetuning on a subset of subjects enable generalization to the rest? We posit that pretraining and finetuning play distinct but complementary roles. During pretraining, the model *acquires factual knowledge*: given a PT subsequence containing s_i, the model learns to decode the attribute type ℓ from the grammar statistics and predict the corresponding attribute a_i^ℓ. During finetuning, the model does not learn new subject-attribute associations; rather, it learns a new *access mechanism:* inferring the attribute type implicitly from the attribute tokens in the context, rather than from explicit grammar cues. Because there is a shared structure between every s_j and its type-ℓ attribute a_j^ℓ, the model can be

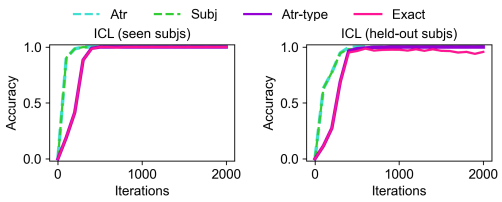

Figure 4: Finetuning the model pretrained on PT sequences, using ICL+Grm sequences with short, variable grammar length with a subset of subjects, leads to good performance on ICL sequences with held-out query subjects.

finetuned on a subset of subjects to learn this implicit inference mechanism and generalize to held-out subjects. We present additional evidence for this mechanism in Section 4 using a simpler setting.

FT on `ICL+Grm` using a subset of subjects. We next ask whether finetuning on `ICL` sequences is necessary to induce implicit attribute-type inference, or whether other distributions can achieve the same effect. To investigate this, we finetune the pretrained model on `ICL+Grm` sequences with short, variable grammar length. Specifically, we use a randomly sampled $S_{\text{ft}} \in \{1, \ldots, 5\}$ to generate each `ICL+Grm` sequence. Due to the short grammar length, the sequence statistics are insufficient to reliably encode the attribute type, encouraging the model to instead infer it implicitly from the attribute tokens in the context. In this sense, these sequences serve the same purpose as `ICL` sequences. However, there remains a format distribution shift: during finetuning, attributes always follow a grammar token, whereas in evaluation `ICL` sequences, attributes directly follow subjects. The model must therefore learn to bridge this format gap at test time. Using variable grammar lengths prevents the model from overfitting to a fixed positional offset, encouraging it to rely on the context to infer the position of the next attribute token. Fig. 4 shows that the model successfully generalizes to `ICL` sequences for both seen and held-out subjects, as it is finetuned on `ICL+Grm` sequences.

Finding 3: Finetuning on `ICL+Grm` sequences with short, variable grammar length using a subset of subjects leads to out-of-distribution generalization on `ICL` sequences with held-out subjects.

We include additional experimental results in Section B. In Fig. 10, we show that finetuning with `ICL+Grm` sequences with longer sequence length ($S_{\text{ft}} = 80$) does not suffice for generalization on `ICL` sequences. In Fig. 11, we show that the sample complexity for finetuning with `ICL` and `ICL+Grm` sequences is comparable. We also examine the effect of changing model scale (layers, heads, embedding dimension) on the results in Fig. 4. The results in Fig. 12 show that the performance remains comparable at moderate scale, but can deteriorate at much smaller or larger scale.

Effect of data distribution. We now investigate how the properties of the pretraining data influence the contextual recall capability. Keeping other parameters fixed, we consider the effect of varying the sequence length S and the diversity between the Markov chains for different attribute types, quantified as $\texttt{Div} := \min_{\ell,\ell'} \|\text{TM}_\ell - \text{TM}_{\ell'}\|_1$, where $\|\cdot\|_1$ denotes the ℓ_1-norm. Intuitively, a higher `Div` for a fixed sequence length implies that the grammar statistics for different attribute types are more distinct, making the attribute type easier to distinguish during pretraining. As shown in Fig. 5, increasing the sequence length S and/or the Markov chain diversity `Div` used while pretraining, improves the model's performance both on the `PT` sequences, as well as on `ICL` sequences with held-out subjects after finetuning on `ICL+Grm` sequences with short, variable grammar length.

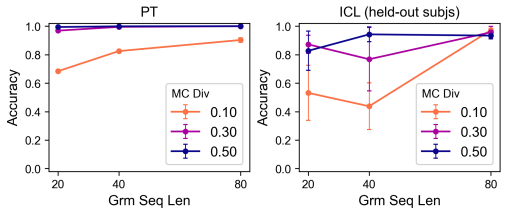

Figure 5: Increasing the sequence length S and/or Markov chain diversity `Div` used while pretraining improves the model's performance on `PT` sequences as well as on `ICL` sequences with held-out subjects after finetuning with `ICL+Grm` sequences.

3.3 Representational Analysis

We now analyze the model's internal representations to study the mechanism underlying contextual recall. Specifically, we investigate whether finetuning on `ICL+Grm` sequences induces the formation of a low-dimensional representation that encodes the shared attribute type from in-context examples.

Let $X_{\ell,t} := [s_{i_0}, a^\ell_{i_0}, [\text{sep}], s_{i_1}, a^\ell_{i_1}, [\text{sep}], \ldots, s_{i_{t+1}}]$ denote an `ICL` sequence for an attribute type ℓ, with $t \in [N]$ demonstrations. For each $\ell \in [L]$, we sample K such sequences, and denote the model's representation at layer j at the last token position as $f_j(X^k_{\ell,t})$ for $k \in [K]$. For fixed j and t, we measure the cosine similarity for inter- and intra-task representations, averaged across the K contexts for each pair $\ell, \ell' \in [L]$. We quantify the representation clustering strength, $\bar{S}^t_j \in [-1, 1]$, in terms of a clustering metric using $1 - \cos(\cdot, \cdot)$ as the distance and attribute-type ℓ as the cluster label (see App. B for details). A high $\bar{S}^t$ indicates that representations of sequences with shared attribute type are tightly clustered and well-separated from those of different attribute types.

Fig. 6 (left) shows the model's performance on ICL sequences with held-out subjects, alongside the representation clustering strength $\bar{S}^t$ computed from layer-2 attention representations, as the number of demonstrations t increases. (Results for other layers are included in App. B.) We consider two settings: Div$\approx$0.2 (top) and Div$\approx$0.5 (bottom). In both cases, accuracy and clustering strength initially improve as t is increased, and eventually saturate. This is also corroborated by the inset figures, which visualize the averaged cosine similarity for inter- and intra-task representations across attribute types, after $t \in \{0, 2, 10\}$ demonstrations. This confirms that the finetuned model aggregates information from multiple examples to form a stable representation of the attribute type.

In Fig. 17 in App. B, we examine the clustering strength using the query subject as the label. We find that layer-1 MLP representations cluster strongly at the last token position. In Section 4, using a simpler setting, we concretely show how query subject and attribute type information is encoded in different model components and combined to succeed on contextual recall.

Here, we note that prior work has proposed two main mechanisms for ICL, namely induction heads and task or function vectors (low-dimensional representations that encode information about the input-output relations in the context) (see App. A for a discussion on related work on task and function vectors for ICL). These results suggest that the formation of these task vectors is the mechanism by which the model succeeds on the contextual recall task in this setting.

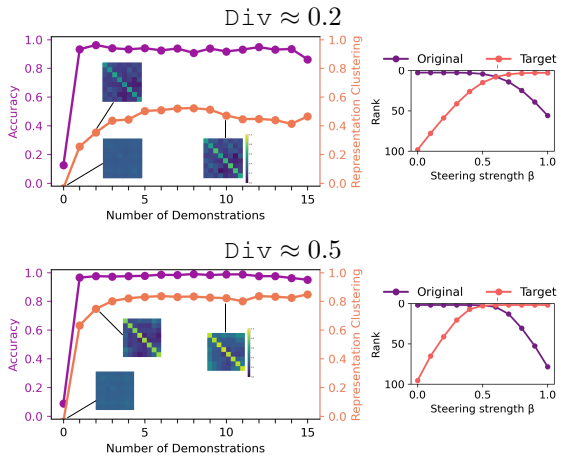

Figure 6: Comparison of accuracy on ICL sequences with held-out subjects, and layer-2 attention representation clustering strength for a finetuned model, as the number of demonstrations is increased (left), and ranks of the original and target attribute type when using those task representations for steering model output (right; see text for details), using Div ≈ 0.2 (top) and Div ≈ 0.5 (bottom). Inset figures visualize the averaged cosine similarity for inter- and intra-task representations. In both cases, both performance and clustering strength improve with number of demonstrations.

Interestingly, while accuracy is comparable in the two settings, higher Div during pretraining leads to stronger representation clustering. This suggests that the separation between the Markov chains (determined by Div) that encode attribute type information during pretraining, is reflected in the task vector separability and strength after finetuning. To corroborate this, we use these representations to steer model outputs. Let $\boldsymbol{v}_\ell := \frac{1}{N}\sum_k f(X^k_{\ell,\mathrm{T}})$ denote a task vector for attribute type ℓ (we use layer-2 attention representations). For some $\beta \in [0, 1]$, we replace $f(X^{k'}_{\ell,\mathrm{T}})$ with $(1-\beta)\boldsymbol{v}_\ell + \beta \boldsymbol{v}_{\ell'}$, and then compare the rank of the original vs. the target attribute based on the model's predicted probabilities. From Fig. 6 (right), we find that as β increases, the model output is steered towards the target. However, for lower Div (top), the required β is much larger compared to the other case.

Finding 4: The finetuned model encodes attribute type information from in-context examples in layer-2 attention representations. Clustering strength increases with both the number of in-context examples and the separation between the attribute-specific Markov chains.

4 Mechanistic Analysis in a Simpler Setting

To gain mechanistic insight into how pretraining and finetuning contribute to contextual recall, we study an analytically tractable setting that preserves the qualitative behavior observed in Findings 1-2 of Section 3.1. We simplify in two ways: i) we encode attribute type information in explicit "relation" tokens rather than grammar sequence statistics, and ii) we use a one-layer attention-only model.

Setup. Let $\mathcal{R} := \{r_1, \ldots, r_L\}$ denote a set of relation tokens, one per attribute type. The PT sequences are generated in a similar manner to Section 2, with a modification to the grammar subsequences. For each subsequence $g_{1:S}$, we first sample a position $i \sim \mathrm{Unif}([S])$ and a relation token $r \sim \mathrm{Unif}(\mathcal{R})$, and set $g_i = r$. Then, we sample the remaining entries independently as $g_{i'} \sim \mathrm{Unif}(\mathcal{G})$ for all $i' \neq i$. The ICL sequences are generated as in Section 2.

We consider a single-layer attention-only model with fixed relative positional encoding added to the key inputs. The output at position t for head $h \in [H]$ and the final model output are defined as:

$$g_h^t(\boldsymbol{X}) = \boldsymbol{X}_{1:t}^\top \boldsymbol{\varphi}((\boldsymbol{X}_{1:t} + \boldsymbol{P}_{\mathrm{T}-t+1:\mathrm{T}})\boldsymbol{W}_{KQ}^h \boldsymbol{x}_t), \quad f^t(\boldsymbol{X}) = \sum_{h=1}^{H} \underbrace{\boldsymbol{W}_{OV}^h g_h^t(\boldsymbol{X})}_{f_h^t(\boldsymbol{X})}, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{\mathrm{T} \times d}$ denotes the input sequence, $\boldsymbol{x}_t \in \mathbb{R}^d$ denotes its tth row, $\boldsymbol{P} = [\boldsymbol{h}(p_{-\mathrm{T}+1}), \ldots, \boldsymbol{h}(p_0)]$ denotes the positional encodings, where $p_{-\mathrm{T}+1}, \ldots, p_0$ denote the position indices, and $\boldsymbol{h}(\cdot) \in \mathbb{R}^d$ denotes the embedding for a vocabulary token or position index. We consider one-hot embeddings and positional encodings and set $d = V + \mathrm{T}$, so that these subspaces are orthogonal. $\boldsymbol{\varphi}(\cdot)$ denotes the softmax. For convenience, we define $\boldsymbol{W}_{OV}^h = (\boldsymbol{W}_O^h)^\top \boldsymbol{W}_V^h$, $\boldsymbol{W}_{KQ}^h = (\boldsymbol{W}_K^h)^\top \boldsymbol{W}_Q^h$, for head h using output,value, key, query weight matrices $\boldsymbol{W}_O^h, \boldsymbol{W}_V^h, \boldsymbol{W}_K^h, \boldsymbol{W}_Q^h \in \mathbb{R}^{d_h \times d}$. We use the shorthand $f(\cdot) = f^{\mathrm{T}}(\cdot)$ and $g_h(\cdot) = g_h^{\mathrm{T}}(\cdot)$ to denote the outputs at the last token position. The model prediction is $v^* = \arg\max_{v \in \mathcal{V}} \boldsymbol{h}(v)^\top f(\boldsymbol{X})$, where we use fixed embeddings and tied unembeddings.

Construction for `PT` Sequences. We first investigate whether an attention-only model is expressive enough to perform factual recall on `PT` sequences of the form

$$\boldsymbol{X} = \boldsymbol{X}_{j_\star}^{\ell_1} || [\boldsymbol{h}(a_{j_\star}^{\ell_1})] || \boldsymbol{X}_{j_\star}^{\ell_2} \ldots \boldsymbol{X}_{j_\star}^{\ell_{N_{\mathrm{tr}}}}, \text{ where } \boldsymbol{X}_{j_\star}^{\ell_1} = [\boldsymbol{h}(s_{j_\star}), \boldsymbol{h}(g_1), ..., \boldsymbol{h}(r_{\ell_1}), ..., \boldsymbol{h}(g_S), \boldsymbol{h}([\text{sep}])]. \tag{2}$$

Here, $\boldsymbol{x}_{\mathrm{T}} = \boldsymbol{h}([\text{sep}])$, $j_\star \sim \mathrm{Unif}([M])$ denotes the subject index, and the correct last-token prediction is $\boldsymbol{h}(a_{j_\star}^{\ell_{N_{\mathrm{tr}}}})$. We first show that there exists an attention-only model capable of perfectly predicting the next token for such sequences. For simplicity, we focus on last-token prediction here; the construction extends to predictions at any position (see App. C for details).

Proposition 1 (Informal)**.** *Consider the input $\boldsymbol{X}$ in Eq. (2). There exists a single-layer attention-only model such that when weights $\boldsymbol{W}_{KQ}^h$ are scaled by a sufficiently large β, it correctly predicts the last token, i.e., the attribute $a_{j_\star}^{\ell_{N_{\mathrm{tr}}}}$ corresponding to the sequence's subject $s_{j_\star}$ and attribute type $\ell_{N_{\mathrm{tr}}}$.*

Proof Sketch. We present a construction with a 3-head model. At a high level, two heads, which we call the **relation** and **subject** heads, are responsible for the prediction at the target position (following $\boldsymbol{h}([\text{sep}])$), and the third **grammar** head for other cases. For simplicity, we present the construction here assuming that the output of the grammar head is zero ($f_{\text{grm}}(\boldsymbol{X}) = 0$), since the outputs from this head do not affect the conclusions in this case, as shown in the full proof in App. C.

The **relation head** attends to the most recent relation token $r_{\ell_{N_{\mathrm{tr}}}}$ and maps it to the sum of *all* attributes of type $\ell_{N_{\mathrm{tr}}}$. Specifically, $\boldsymbol{W}_{KQ}^{\text{rel}} = \beta(\sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{p})\boldsymbol{h}([\text{sep}])^\top$, $\boldsymbol{W}_{OV}^{\text{rel}} = \sum_\ell (\sum_j \boldsymbol{h}(u_j^\ell))\boldsymbol{h}(r_\ell)^\top$, where $\boldsymbol{p} := \sum_{i=1}^{S+2} \boldsymbol{h}(p_{-i+1})$. With $\beta \to \infty$, the head's outputs become $g_{\text{rel}}(\boldsymbol{X}) = \boldsymbol{h}(r_{\ell_{N_{\mathrm{tr}}}})$, *i.e.*, the most recent relation token in the sequence, and $f_{\text{rel}}(\boldsymbol{X}) = \sum_j \boldsymbol{h}(u_j^{\ell_{N_{\mathrm{tr}}}})$, *i.e.*, attributes of type $\ell_{N_{\mathrm{tr}}}$.

The **subject head** attends to the subject token $s_{j_\star}$ and filters out attributes not associated with it. Specifically, $\boldsymbol{W}_{KQ}^{\text{subj}} = \beta(\sum_j \boldsymbol{h}(s_j))\boldsymbol{h}([\text{sep}])^\top$, $\boldsymbol{W}_{OV}^{\text{subj}} = -\sum_j (\sum_{j' \neq j, \ell} \boldsymbol{h}(a_{j'}^\ell))\boldsymbol{h}(s_j)^\top$. With $\beta \to \infty$, this head outputs $g_{\text{subj}}(\boldsymbol{X}) = \boldsymbol{h}(s_{j_\star})$, the subject token in the sequence, and $f_{\text{subj}}(\boldsymbol{X}) = -(\sum_u \boldsymbol{h}(u) - \sum_\ell \boldsymbol{h}(a_{j_\star}^\ell))$, *i.e.*, the negative of attributes not associated with subject $s_{j_\star}$.

Combining these outputs, the model isolates the specific attribute $a_{j_\star}^{\ell_{N_{\mathrm{tr}}}}$, *i.e.*, the attribute reinforced by both heads: $f(\boldsymbol{X}) = \sum_j \boldsymbol{h}(u_j^{\ell_{N_{\mathrm{tr}}}}) + \sum_\ell \boldsymbol{h}(a_{j_\star}^\ell) - \sum_u \boldsymbol{h}(u)$, yielding the correct prediction $v^* = a_{j_\star}^{\ell_{N_{\mathrm{tr}}}}$.

Construction for `ICL` Sequences. Next, consider `ICL` sequences of the form

$$\boldsymbol{X} = [\boldsymbol{h}(s_{j_1}), \boldsymbol{h}([\text{sep}]), \boldsymbol{h}(a_{j_1}^{\ell_\star}), ..., \boldsymbol{h}(s_{j_{N_{\text{ft}}+1}}), \boldsymbol{h}([\text{sep}])], \tag{3}$$

with correct last token prediction $\boldsymbol{h}(a_{j_{N_{\text{ft}}+1}}^{\ell_\star})$, where $\ell_\star \sim \mathrm{Unif}([L])$ denotes the shared attribute type. We show that there exists an attention-only model that perfectly predicts the last token.

Proposition 2 (Informal)**.** *Consider the input $\boldsymbol{X}$ in Eq. (3). There exists single-layer attention-only model such that when weights $\boldsymbol{W}_{KQ}^h$ are scaled by a sufficiently large β, it correctly predicts the last token, i.e., attribute $a_{j_{N_{\text{ft}}+1}}^{\ell_\star}$ corresponding to the query subject $s_{j_{N_{\text{ft}}+1}}$ and the shared attribute type $\ell_\star$.*

Proof Sketch. We adapt the construction from Proposition 1 with minimal changes, but with one crucial modification that reflects the role of finetuning. The **subject** head operates similarly, attending to the query subject and filtering out attributes not associated with it. The key difference lies in the **relation** head: since ICL sequences contain no explicit relation tokens, this head must now *infer the attribute type implicitly* by attending to the attribute tokens in the context and mapping them to all attributes of the same type $\ell_\star$. Specifically, for the **subject** head, we set

$$\boldsymbol{W}_{KQ}^{\text{subj}} = \beta\big(\sum_j \boldsymbol{h}(s_j) + \boldsymbol{h}(p_{-1})\big)\boldsymbol{h}([\text{sep}])^\top, \quad \boldsymbol{W}_{OV}^{\text{subj}} = -\sum_j \big(\sum_{j'\neq j,\ell} \boldsymbol{h}(a_{j'}^{\ell})\big)\boldsymbol{h}(s_j)^\top. \tag{4}$$

Compared to the construction for PT sequences (Prop. 1), $\boldsymbol{W}_{OV}^{\text{subj}}$, which contains subject-attribute information, is unchanged. The only difference is that $\boldsymbol{W}_{KQ}^{\text{subj}}$ now contains $\boldsymbol{h}(p_{-1})$, so that when $\beta\to\infty$, $g_{\text{subj}}(\boldsymbol{X}) = \boldsymbol{h}(s_{j_{N_{\text{ft}}+1}})$, *i.e.*, the query subject, and hence, $f_{\text{subj}}(\boldsymbol{X}) = -(\sum_u \boldsymbol{h}(u) - \sum_\ell \boldsymbol{h}(a^\ell_{j_{N_{\text{ft}}+1}}))$.

We now discuss the **relation** head. Since in our experiments, we finetune with a subset of subjects, let $\mathcal{S}' \subset \mathcal{S}$ denote a subset of subjects. Next, for each attribute type ℓ, let $\mathcal{U}'_\ell := \cup_{j\in\mathcal{S}'}\{a_j^\ell\}$ denote the set of unique attributes seen during finetuning. With this notation established, we set

$$\boldsymbol{W}_{KQ}^{\text{rel}} = \beta\sum_\ell \big(\boldsymbol{h}(r_\ell) + \boldsymbol{u}'_\ell\big)\boldsymbol{h}([\text{sep}])^\top, \quad \boldsymbol{W}_{OV}^{\text{rel}} = \sum_\ell \big(\big(\sum_{u\in\mathcal{U}_\ell}\boldsymbol{h}(u)\big)\boldsymbol{h}(r_\ell)^\top + \boldsymbol{u}'_\ell\boldsymbol{u}_\ell'^\top\big), \tag{5}$$

where $\boldsymbol{u}'_\ell := \sum_{u\in\mathcal{U}'_\ell}\boldsymbol{h}(u)$. In contrast to the construction for PT sequences, where both $\boldsymbol{W}_{KQ}^{\text{rel}}$ and $\boldsymbol{W}_{OV}^{\text{rel}}$ rely on the relation tokens $\boldsymbol{h}(r_\ell)$, in this case, they rely on the attributes seen during finetuning, *i.e.*, $\boldsymbol{u}'_\ell$. When $\beta\to\infty$, $g_{\text{rel}}(\boldsymbol{X}) = \frac{1}{N_{\text{ft}}}\sum_i \boldsymbol{h}(a_{j_i}^{\ell_\star})$, *i.e.*, the average of the attributes that appear in the context, and $f_{\text{rel}}(\boldsymbol{X}) = \sum_j \boldsymbol{h}(u_j^{\ell_\star})$, retrieving the sum of all attributes of the shared type $\ell_\star$.

Finally, by combining the outputs from the two heads, the model isolates the specific attribute encouraged by both and gives final prediction $v^* = a^{\ell_\star}_{j_{N_{\text{ft}}+1}}$.

Note that, for the **relation** head to output all attributes of the same type as the *attributes* that appear in the context, for any type ℓ, the set of attributes seen during finetuning $\mathcal{U}'_\ell$ should match the full set $\mathcal{U}_\ell$. However, since attributes are shared among subjects, the set of subjects seen during finetuning can be much smaller than the full set. This helps explain why finetuning on a subset of subjects can enable generalization on held-out subjects.

Training Dynamics for Finetuning. In this section, we consider a 1-layer 2-head attention-only model initialized with the construction for PT sequences in Proposition 1, and examine its training dynamics for finetuning on ICL sequences with the last-token prediction objective (using a subset of subjects, but with all attributes seen during FT). We show that after updating the weights with one or two steps of GD, under certain assumptions on the data generation process parameters, the model can attain perfect accuracy on ICL sequences with held-out query subjects.

Theorem 1 (Informal)**.** *Consider a 1-layer 2-head attention-only model defined in Eq. (1), initialized with the weights from the construction for PT sequences in Proposition 1. Assuming sufficiently large M, M_{ft}, small N_{ft}, and $\mathcal{U}'_\ell = \mathcal{U}_\ell, \forall \ell \in [L]$, after one step of GD on $\boldsymbol{W}_{KQ}^{subj}$, one step of projected GD on $\boldsymbol{W}_{OV}^{rel}$, and two steps of GD on $\boldsymbol{W}_{KQ}^{rel}$, for last-token prediction on the population of ICL sequences with a subset of M_{ft} subjects, the updated weights satisfy*

$$\boldsymbol{W}_{KQ}^{subj} \approx \beta\Big(\sum_j \boldsymbol{h}(s_j) + \lambda_1\boldsymbol{h}(p_{-1})\Big)\boldsymbol{h}([sep])^\top, \tag{6}$$

$$\boldsymbol{W}_{KQ}^{rel} \approx \beta\Big(\sum_\ell\Big(\boldsymbol{h}(r_\ell) + \tfrac{2}{3}\tfrac{\lambda_2}{LU_L}\boldsymbol{u}_\ell\Big) + \tfrac{\lambda_2}{N_{ft}}\sum_{i=1}^{N_{ft}}\boldsymbol{h}(p_{1-3i}) - \lambda_2\Big(\tfrac{1}{\mathrm{T}}\boldsymbol{p} + \tfrac{1}{3}\tfrac{1}{M}\sum_s \boldsymbol{h}(s)\Big)\Big)\boldsymbol{h}([sep])^\top,$$

$$\boldsymbol{W}_{OV}^{rel} \approx \sum_\ell \boldsymbol{u}_\ell\boldsymbol{h}(r_\ell)^\top + \lambda_3 N_{ft}\sum_\ell \boldsymbol{u}_\ell\boldsymbol{u}_\ell^\top + \lambda_3\tfrac{U_L^2}{M_{ft}}\sum_{j=1}^{M_{ft}}\Big(\sum_\ell \boldsymbol{h}(a_j^\ell)\Big)\boldsymbol{h}(s_j)^\top, \quad \boldsymbol{u}_\ell := \sum_{u\in\mathcal{U}_\ell}\boldsymbol{h}(u), \tag{7}$$

where $\lambda_1, \lambda_2, \lambda_3$ are some constants. These weights, with sufficiently large β, lead the model to attain perfect accuracy on ICL sequences for all subjects.

We defer the full statement and its proof to App. C. Interestingly (albeit under certain assumptions), the updated weights after finetuning recover the construction in Proposition 2 (upto some additive factors that don't change the model functionality), as we discuss next.

Subject head. Comparing $\boldsymbol{W}_{KQ}^{\text{subj}}$ in Eq. (6) with Eq. (4), the only difference is the second term is scaled by λ_1 ($\propto \eta_{KQ}^{\text{subj}}$, the step-size). When $\beta \to \infty$, regardless of this coefficient, the attention concentrates on the query subject $s_{j_{N_{\text{ft}}+1}}$. Crucially, $\boldsymbol{W}_{OV}^{\text{subj}}$ does not need to be updated since the subject-attribute associations are already learned during pretraining.

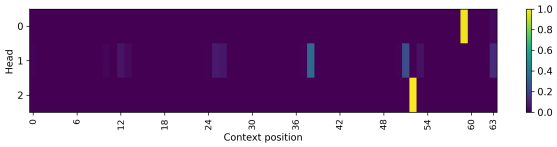

Figure 7: Attention scores for each head across a PT sequence at the end of pretraining. Head 2 attends to the most recent subject, while head 0 attends to the most recent relation token, as predicted by Prop. 1.

Relation head. Next, for $\boldsymbol{W}_{KQ}^{\text{rel}}$, comparing the three terms in Eq. (7) with Eq. (5): the first term matches the construction (other than the coefficient $\frac{2\lambda_2}{3LU_L}$), while the second contains positional encodings for the positions at which attributes appear; these terms boost the attention on the attribute tokens in the context. The third term contains i) a positional encoding term (which contributes uniformly to every token in the context, leaving the softmax scores unchanged), and ii) a subject token term that suppresses the attention at the subject tokens. For $\boldsymbol{W}_{OV}^{\text{rel}}$, the first two terms in Eq. (7) match the construction (other than the coefficient $\lambda_3 N_{\text{ft}}$). The third term encodes subject-attribute associations for the finetuning subjects. However, since the KQ part for this head isolates attribute tokens, this term does not contribute to the output.

Experimental Validation. We present experimental evidence to corroborate our theoretical constructions, using a 1-layer 3-head attention-only model (see App. B for details). We first train the model using PT sequences (Eq. (2)) with the next-token prediction objective. In Fig. 7, we visualize the attention scores for each head across a PT sequence, and find that the heads specialize into distinct roles consistent with Prop. 1. We find that head 2 attention score concentrates on the most recent subject token, and it outputs $\boldsymbol{h}(s_{j_\star})$, matching the role of $g_{\text{subj}}(\boldsymbol{X})$ in our construction, while head 0 attends to the most recent relation token, *i.e.*, it outputs $\boldsymbol{h}(r_{\ell_{N_{\text{tr}}}})$, consistent with $g_{\text{rel}}(\boldsymbol{X})$. Further, in Fig. 8, we compute the cosine similarity between the head outputs and theoretical outputs of the subject and relation heads specified by our construction. We find that head 2 output closely matches $f_{\text{subj}}(\boldsymbol{X})$, while head 0 matches $f_{\text{rel}}(\boldsymbol{X})$. Similarly, in Figs. 19 and 20 in App. C, we present experimental evidence to corroborate the construction for ICL sequences in Prop. 2.

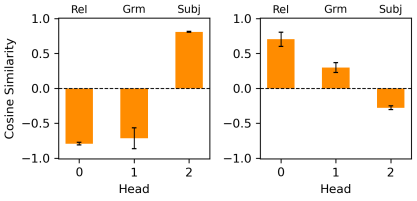

Figure 8: Cosine similarity between the actual outputs of each head and the outputs $f_{\text{subj}}(\boldsymbol{X})$ (left) and $f_{\text{rel}}(\boldsymbol{X})$ (right) from our construction in Proposition 1. Head 2 output closely matches $f_{\text{subj}}(\boldsymbol{X})$, while head 0 matches $f_{\text{rel}}(\boldsymbol{X})$.

Effect of Number of Attributes seen during Finetuning. Based on our construction for ICL sequences in Prop. 2, we note that generalization on held-out query subjects depends on how many unique attributes are seen during FT. To validate this effect, in Fig. 23 in the App., we find that for a fixed number of subjects used during finetuning (32 or 64), increasing the number of unique attributes seen during finetuning improves the model's performance on ICL sequences with held-out query subjects.

5 Conclusion

We studied contextual recall, a form of ICL that requires models pretrained to acquire knowledge about various subjects and associated attributes, to recall a specific attribute for a query subject by implicitly inferring the relevant attribute type based on in-context examples. Our results give insights into the complementary roles of pretraining and finetuning in enabling in-context reasoning involving learned knowledge. We believe that our synthetic data setup can be a useful testbed to study several questions. This includes examining the effects of using a mixture of pretraining and finetuning data instead of using them in a stage-wise manner, curriculum learning, non-uniform distributions over the subjects instead of uniform as we use, increasing the number of attribute types, and so on. A key question is to better understand when and how can the representations learned during pretraining be improved to achieve better generalization after finetuning.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods and assist in the writing of proofs. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology or experiments, or interpret results, and other cases to assist with translation, clean and reformat dataset, support qualitative and thematic data analysis, are not applicable to this work. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? Investigations with linear models. In *Int. Conference on Learning Representations*, 2023.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. *arXiv preprint arXiv:2309.14316*, 2023.

Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Tina Behnia, Puneesh Deora, and Christos Thrampoulidis. Facts in stats: Impacts of pretraining diversity on language model generalization. *arXiv preprint arXiv:2510.16096*, 2025.

Satwik Bhattamishra, Arkil Patel, Phil Blunsom, and Varun Kanade. Understanding in-context learning in transformers and LLMs by learning to learn discrete functions. In *The Twelfth International Conference on Learning Representations*, 2024.

Alberto Bietti, Vivien Cabannes, Diane Bouchacourt, Herve Jegou, and Leon Bottou. Birth of a transformer: A memory viewpoint. *Advances in Neural Information Processing Systems*, 36: 1560–1588, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020.

Liam Carroll, Jesse Hoogland, Matthew Farrugia-Roberts, and Daniel Murfet. Dynamics of transient structure in in-context linear regression transformers. *arXiv preprint arXiv:2501.17745*, 2025.

Stephanie C.Y. Chan, Adam Santoro, Andrew K. Lampinen, Jane X. Wang, Aaditya K. Singh, Pierre H. Richemond, James L. McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Lei Chen, Joan Bruna, and Alberto Bietti. Distributional associations vs in-context reasoning: A study of feed-forward and attention layers. In *The Thirteenth International Conference on Learning Representations*, 2025.

Siyu Chen, Heejune Sheen, Tianhao Wang, and Zhuoran Yang. Training dynamics of multi-head softmax attention for in-context learning: Emergence, convergence, and optimality. *arXiv preprint arXiv:2402.19442*, 2024a.

Siyu Chen, Heejune Sheen, Tianhao Wang, and Zhuoran Yang. Unveiling induction heads: Provable training dynamics and feature learning in transformers. *arXiv preprint arXiv:2409.10559*, 2024b.

Francesco D'Angelo, Francesco Croce, and Nicolas Flammarion. Selective induction heads: How transformers select causal structures in context. In *The Thirteenth International Conference on Learning Representations*, 2025.

Puneesh Deora, Bhavya Vasudeva, Tina Behnia, and Christos Thrampoulidis. In-context occam's razor: How transformers prefer simpler hypotheses on the fly. In *Second Conference on Language Modeling*, 2025.

Yuxin Dong, Jiachen Jiang, Zhihui Zhu, and Xia Ning. Understanding task vectors in in-context learning: Emergence, functionality, and limitations. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=CLBVilFk7N.

Ezra Edelman, Nikolaos Tsilivis, Benjamin Edelman, Eran Malach, and Surbhi Goel. The evolution of statistical induction heads: In-context learning markov chains. *Advances in Neural Information Processing Systems*, 37:64273–64311, 2024a.

Ezra Edelman, Nikolaos Tsilivis, Benjamin L. Edelman, Eran Malach, and Surbhi Goel. The evolution of statistical induction heads: In-context learning markov chains. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b.

Deqing Fu, Tian-Qi Chen, Robin Jia, and Vatsal Sharan. Transformers learn to achieve second-order convergence rates for in-context linear regression. *Advances in Neural Information Processing Systems*, 37:98675–98716, 2024.

Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Seungwook Han, Jinyeop Song, Jeff Gore, and Pulkit Agrawal. Emergence and effectiveness of task vectors in in-context learning: An encoder decoder perspective. In *Forty-second International Conference on Machine Learning*, 2025.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. *arXiv preprint arXiv:2310.15916*, 2023.

Guan Zhe Hong, Bhavya Vasudeva, Vatsal Sharan, Cyrus Rashtchian, Prabhakar Raghavan, and Rina Panigrahy. Latent concept disentanglement in transformer-based language models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Andrej Karpathy. mingpt. https://github.com/karpathy/minGPT?tab=readme-ov-file [Accessed: March 4, 2024].

Ryotaro Kawata, Yujin Song, Alberto Bietti, Naoki Nishikawa, Taiji Suzuki, Samuel Vaiter, and Denny Wu. From shortcut to induction head: How data diversity shapes algorithm selection in transformers. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Ziqian Lin and Kangwook Lee. Dual operating modes of in-context learning. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Sheng Liu, Haotian Ye, Lei Xing, and James Zou. In-context vectors: making in context learning more effective and controllable through latent space steering. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2019.

Eshaan Nichani, Alex Damian, and Jason D. Lee. How transformers learn causal structure with gradient descent. In *Forty-first International Conference on Machine Learning*, 2024a.

Eshaan Nichani, Jason D Lee, and Alberto Bietti. Understanding factual recall in transformers via associative memories. *arXiv preprint arXiv:2412.06538*, 2024b.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8298–8319, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.527.

Core Francisco Park, Ekdeep Singh Lubana, and Hidenori Tanaka. Algorithmic phases of in-context learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Nived Rajaraman, Marco Bondaschi, Ashok Vardhan Makkuva, Kannan Ramchandran, and Michael Gastpar. Transformers on markov data: Constant depth suffices. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Allan Raventos, Mansheej Paul, Feng Chen, and Surya Ganguli. Pretraining task diversity and the emergence of non-bayesian in-context learning for regression. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Aaditya K. Singh, Stephanie C. Y. Chan, Ted Moskovitz, Erin Grant, Andrew M. Saxe, and Felix Hill. The transient nature of emergent in-context learning in transformers. *arXiv preprint arXiv:2311.08360*, 2023.

Aaditya K Singh, Ted Moskovitz, Felix Hill, Stephanie CY Chan, and Andrew M Saxe. What needs to go right for an induction head? a mechanistic study of in-context learning circuits and their formation. In *International Conference on Machine Learning*, pp. 45637–45662. PMLR, 2024.

Aaditya K. Singh, Ted Moskovitz, Sara Dragutinovic, Felix Hill, Stephanie C. Y. Chan, and Andrew M. Saxe. Strategy coopetition explains the emergence and transience of in-context learning. *arXiv preprint arXiv:2503.05631*, 2025.

Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. *arXiv preprint arXiv:2212.07677*, 2022.

Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, and Peter L Bartlett. How many pretraining tasks are needed for in-context learning of linear regression? *arXiv preprint arXiv:2310.08391*, 2023.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*, 2022.

Liu Yang, Ziqian Lin, Kangwook Lee, Dimitris Papailiopoulos, and Robert Nowak. Task vectors in in-context learning: Emergence, formation, and benefit. *arXiv preprint arXiv:2501.09240*, 2025.

Kayo Yin and Jacob Steinhardt. Which attention heads matter for in-context learning? In *Forty-second International Conference on Machine Learning*, 2025.

Ruiqi Zhang, Spencer Frei, and Peter L Bartlett. Trained transformers learn linear models in-context. *arXiv preprint arXiv:2306.09927*, 2023.

Ruiqi Zhang, Jingfeng Wu, and Peter Bartlett. In-context learning of a linear transformer block: Benefits of the mlp component and one-step gd initialization. *Advances in Neural Information Processing Systems*, 37:18310–18361, 2024.

Yedi Zhang, Aaditya K. Singh, Peter E. Latham, and Andrew Saxe. Training dynamics of in-context learning in linear attention. *arXiv preprint arXiv:2501.16265*, 2025.

Nicolas Zucchet, Jörg Bornschein, Stephanie Chan, Andrew Lampinen, Razvan Pascanu, and Soham De. How do language models learn facts? dynamics, curricula and hallucinations. *arXiv preprint arXiv:2503.21676*, 2025.

APPENDIX

A Related Work **14**

B Additional Experimental Results and Details of Experimental Settings **16**

B.1 Details of Experimental Settings 16

B.2 Additional Experimental Results 17

B.3 Details of Representational Analysis 18

C Analysis for the Attention-Only Model **20**

C.1 Constructions for PT and ICL Sequences 20

C.2 Optimization Dynamics for Finetuning 22

C.3 Experimental Validation 29

D Limitations **32**

A RELATED WORK

In this section, we discuss related work on studying ICL and factual recall in transformers using controlled/synthetic settings, as well as on task vectors for ICL.

Factual Recall. Recent studies have utilized controlled synthetic setups to systematically explore how transformers acquire and recall factual associations. Allen-Zhu & Li (2023) and Zucchet et al. (2025) employ synthetic biography datasets, where each entry contains information about an individual or subject in the form of multi-sentence paragraphs. Each sentence associates some fact or attribute with an individual or subject, via a sentence template that contains information about the relation or attribute type, such as a person's birthplace, in the linguistic structure. They consider a fixed set of templates for each attribute type. Zucchet et al. (2025) evaluate factual knowledge (*i.e.*, correct attribute token prediction) using similar biography entries that are generated with templates from a held-out set, and identify a stage-wise learning dynamic where the model first learns to predict some attribute token, and then the correct attribute. We also observe similar stage-wise learning in our setup during pretraining in Section 3.1. In Allen-Zhu & Li (2023), the model is evaluated on question-answering (QA) formats, which constitutes a distribution shift from the biography-style format. Similar to our results in Section 3.2, they find that while pretraining is insufficient, finetuning (on QA) with a subset of subjects enables generalization on held-out subjects. Crucially, however, their QA prompts still contain explicit relation or attribute type information (e.g., "What is [subject]'s city of birth?"), whereas our contextual recall task constitutes a more substantial distribution shift as the context does not contain explicit relation cues. Investigating whether we see similar results as in Section 3.2 for our synthetic setup with synthetic biography data used in these studies would be an interesting direction for future work.

In addition to these works, other studies have adopted abstracted setups with subject-grammar-attribute triplets to study factual recall (Nichani et al., 2024b; Behnia et al., 2025). Nichani et al. (2024b) uses a data setup where the relation (or attribute type) is a single, dedicated token, while the rest of the tokens in the grammar sequence are sampled randomly. Using this setup, they show that a one-layer transformer model (with a self-attention layer followed by an MLP) succeeds on factual recall whenever either the total number of self-attention parameters or MLP parameters scales (up to log factors) linearly with the number of facts. In contrast, Behnia et al. (2025) propose a setup to more faithfully capture the statistical structure in the grammar sequences, *i.e.*, templates in the controlled language datasets in Zucchet et al. (2025); Allen-Zhu & Li (2023). They model each template as a randomly sampled Markov chain, and omit the relation token (the facts are independent of the grammar), to study how sequence statistics affect generalization. Interestingly, they show that several phenomena observed with the controlled language datasets in Zucchet et al. (2025); Allen-Zhu & Li (2023) can be reproduced in this abstracted setup.

A key differentiator across these frameworks is how "relation" (or attribute type) information is represented. While in Nichani et al. (2024b) the relation is a specific token, Behnia et al. (2025) omits the relation token entirely. Our synthetic framework bridges these approaches: similar to Behnia et al. (2025), we model templates using Markov chains, but in contrast to their work, and similar to the other prior studies, we retain the relation information by associating the Markov chains with specific attribute types.

For mechanistic analysis, we adopt an abstracted setup similar to Nichani et al. (2024b) and compare our attention-only construction for factual recall (Proposition 1) with theirs. In both constructions, the relation heads perform a similar role: they attend to the (most recent) relation token in the context to output the sum of all attributes of the relevant type. However, the subject heads behave differently. While the construction in Nichani et al. (2024b) uses subject heads to boost the attributes associated with the subject present in the context, our construction utilizes subject heads to suppress all attributes not associated with the subject. While we present experimental validation for our construction, we note that changing the initialization seed can lead to the model using a different mechanism, similar to their construction.

As discussed in Section 1, while prior works focus on factual recall using explicit cues about the attribute type from the context, our work investigates contextual recall, a more challenging task that requires the model to infer the attribute type implicitly from in-context examples.

In-Context Learning. Several recent studies have leveraged controlled synthetic environments to analyze how Transformers learn in-context when trained from scratch. A common approach involves training models on well-defined function classes, most notably linear regression (Garg et al., 2022; Raventos et al., 2023; Akyürek et al., 2023; von Oswald et al., 2022; Ahn et al., 2023; Wu et al., 2023) and Markov chains (Edelman et al., 2024b; Park et al., 2025; Rajaraman et al., 2024; Deora et al., 2025). These works often explore whether Transformers implement specific algorithms or functionalities—such as gradient descent (von Oswald et al., 2022; Ahn et al., 2023; Bai et al., 2023) or higher-order-algorithms (Fu et al., 2024) for linear regression, and induction heads for copying tasks (Olsson et al., 2022), Markov chain-based setups (Edelman et al., 2024b; Rajaraman et al., 2024; Chen et al., 2024b; D'Angelo et al., 2025), settings that combine these (Bietti et al., 2023; Chen et al., 2025; Kawata et al., 2025), as well as other controlled synthetic settings (Singh et al., 2024). Further research (Raventos et al., 2023; Park et al., 2025; Lin & Lee, 2024; Singh et al., 2025) has investigated task diversity, comparing *task retrieval* and *task learning* modes of ICL (Pan et al., 2023). Additionally, research has explored transient dynamics to understand how these two modes evolve over the course of training (Singh et al., 2023; Carroll et al., 2025; Singh et al., 2025). Finally, a growing body of work also explores the training dynamics of in-context learning by examining the optimization dynamics of linear regression (Zhang et al., 2025; 2023; 2024) in both one-layer linear attention and softmax attention models (Chen et al., 2024a), as well as for learning causal structures Nichani et al. (2024a).

Chan et al. (2022) study how properties of the training distribution determine whether a transformer adopts an ICL or an in-weights learning (IWL) strategy. In their ICL evaluation, the context supplies image-label pairs with episode-specific label assignments, so the query can be answered entirely from the associations presented in context. Their IWL evaluation instead tests retrieval of fixed image-label associations learned during training. Our task is fundamentally different. Contextual recall requires the model to combine both capabilities within a single prediction: it must use the context to infer a latent attribute type, and then retrieve the corresponding subject-specific fact from knowledge stored during PT. Since each subject is associated with multiple attributes, the context does not directly provide the answer. It only identifies which of the subject's stored attributes is relevant.

A defining characteristic of the aforementioned studies is that the training and inference formats are identical; the model is evaluated on the same sequence structures it encountered during training. In contrast, our work on contextual recall introduces a significant prompt-distribution shift. While pretraining focuses on instilling factual knowledge in the model via explicit grammar-based cues, the ICL evaluation requires the model to transition a novel format necessitating inference of the relevant attribute type implicitly from the in-context demonstrations. Therefore, unlike standard ICL setups that focus on function induction, our framework requires the model to bridge the gap between structured knowledge acquisition and implicit in-context reasoning, as mentioned in Section 1.

In contrast to our work, Xie et al. (2022) investigate a distribution shift that is primarily compositional rather than structural. In their framework, the shift occurs when a model pretrained on long, continuous documents is prompted with a sequence of independent, i.i.d. examples. While they frame ICL as a statistical process of implicit Bayesian inference, our work provides a mechanistic perspective discussed in Section 3.3 and Section 4.

Task and Function Vectors in ICL. Recent work studying ICL mechanisms in pretrained LLMs shows that certain directions in activation space encode the input–output relationship of the ICL task (Hendel et al., 2023; Todd et al., 2024; Yin & Steinhardt, 2025). These activation directions, referred to as task or function vectors, seem to encode abstract task information that is largely independent of the specific examples in the prompt, and enable causal interventions that modify model behavior (Liu et al., 2024). Other works use controlled synthetic setups to study the emergence of task vectors for ICL of some function classes (Yang et al., 2025; Dong et al., 2026), how task vector encoding is correlated with task-level decoding (Han et al., 2025), as well as task vector disentanglement and composition for two-hop reasoning tasks, and geometry for tasks with numerical-valued continuous latent parameters (Hong et al., 2026). Consistent with prior work, our analysis in Section 3.3 suggests that finetuning drives the emergence of low-dimensional task vectors for the shared attribute type, which enables the model to do well on the contextual recall task. Beyond observing their existence, we make two additional contributions. First, we show that the separability and functional utility of these task vectors (measured through representation clustering and output steering) is directly governed by properties of the pretraining data, specifically the separation between attribute-specific Markov chains. Second, in Section 4, we provide a mechanistic explanation for how task vectors emerge through finetuning: the relation head's OV matrix learns attribute-to-attribute associations for shared attribute type, which induces the clustering of representations by attribute type.

B Additional Experimental Results and Details of Experimental Settings

B.1 Details of Experimental Settings

All experiments were conducted on an internal cluster with a V100 GPU.

We use a 2-layer 1-head GPT-2 type decoder-only transformer model (Karpathy) with embedding dimension 256. We train the model with AdamW optimizer (Loshchilov & Hutter, 2019) with learning rate 10^{-4}, weight decay 0.001, and batch size 64, for both pretraining and finetuning.

For evaluation, we report four metrics: *Exact Match* (predicted attribute matches the ground truth), *Attribute-Type Match* (predicted attribute has the correct type), *Subject Match* (predicted attribute belongs to the correct subject), and *Attribute Match* (any attribute token is predicted). See Fig. 1 for an illustration.

For the experiments in Section 3.1, we set $M = 256, L = 8, N_{\text{tr}} = 5, M_1 = 256, M_2 = \cdots = M_8 = 32, S = 80$. We set the grammar-only subsequence probability $p_G = 0.2$, and separation between Markov chains $\texttt{Div} \approx 0.5$. To control for $\texttt{Div}$, we first randomly generate a large pool of transition matrices, and then use greedy selection to curate a subset of transition matrices that are assigned to each attribute type. We pretrain the model for $20k$ iterations.

For the experiments in Section 3.2, we use the same pretraining setting as in Section 3.1, and set $N = N_{\text{ft}} = 16$. We use 128 subjects for finetuning, unless stated otherwise. For experiments with $\texttt{ICL+Grm}$ sequences, $S_{\text{ft}} \sim \text{Unif}(\{1, \ldots, 5\})$. In all cases, we finetune for $2k$ iterations. In Fig. 11, we compare the best performance across finetuning iterations.

For Figs. 5, 11 and 13, the results are reported after averaging across two random initialization seeds.

For the other experiments in Figs. 5 and 13, the details are as follows. All results are reported at end of pretraining/finetuning. We consider $\texttt{Div} \approx 0.1, 0.3, 0.5$. In Fig. 5, we use fixed $M = 256$ and for $S = 20$, we use $S_{\text{ft}} \sim \text{Unif}(\{1, ..., 4\})$, while for $S = 40$ or $S = 80$, we use $S_{\text{ft}} \sim \text{Unif}(\{1, ..., 5\})$. In Fig. 13, we use fixed $S = 80$, $S_{\text{ft}} \sim \text{Unif}(\{1, ..., 5\})$.

The results in Fig. 6 (bottom) are reported with the same setting as Fig. 4. For the top plot, we only change $\texttt{Div} \approx 0.2$. We use 50 sequences for each attribute type.

The settings for the experiments in Section 4 and App. C are as follows. We set $M = 256, L = 8, N_{\text{tr}} = 5, M_2 = \cdots = M_8 = 32, S = 10, N = N_{\text{ft}} = 5$ for both sections, while $M_1 = 32$ and $M_1 = 256$, respectively. We train an attention-only model with $d_h = 256$ using AdamW optimizer with learning rate 0.001, weight decay 0.001, and batch size 64, for both pretraining and finetuning. We use a cosine learning rate scheduler for pretraining. The results with $M = 2048$ subjects use the same experimental setting, except we set $d_h = 2048$. Unless stated otherwise, we pretrain for $5k$ iterations and finetune for $2k$ iterations. For the results for experimental validation in Section 4, we pretrain for $2k$ iterations and finetune for $1k$ iterations.

B.2 Additional Experimental Results

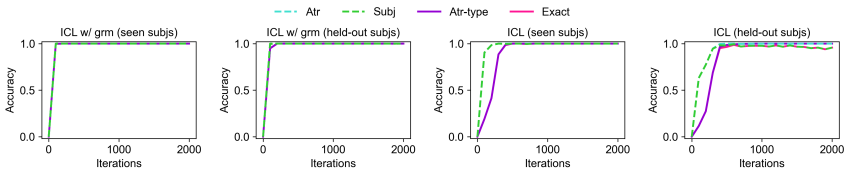

Figure 9: Performance of the model when finetuning with `ICL+Grm` sequences with a subset of subjects (same setting as Fig. 4), on `ICL+Grm` sequences with $S_{\text{ft}} = 1$ (left) and `ICL` sequences (right) with seen or held-out subjects. Finetuning with `ICL+Grm` sequences enables out-of-distribution generalization on `ICL` sequences with held-out subjects. Performance improves first on `ICL+Grm` and later on `ICL` sequences.

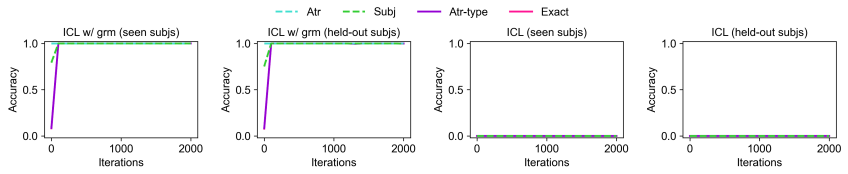

Figure 10: Performance of the model when finetuning with `ICL+Grm` sequences with $S_{\text{ft}} = 80$ with a subset of subjects, on `ICL+Grm` sequences (left) and `ICL` sequences (right) with seen or held-out subjects. Finetuning with `ICL+Grm` sequences with long grammar sequence length does not enable out-of-distribution generalization on `ICL` sequences with held-out subjects.

Sample Complexity. We investigate whether the format distribution shift inherent in finetuning with `ICL+Grm` sequences incurs a cost in terms of sample efficiency compared to direct finetuning with `ICL` sequences. In Fig. 11, we compare the `ICL` performance on held-out subjects as we vary the number of finetuning subjects. Interestingly, we find that the sample complexity is comparable across both settings, with performance improving monotonically as the number of finetuning subjects increases.

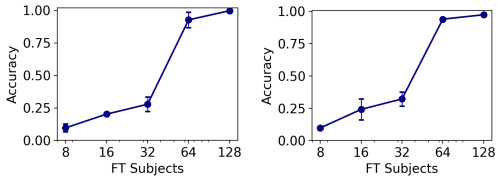

Figure 11: Increasing the number of finetuning subjects with `ICL` (left) or `ICL+Grm` (right) sequences improves the model's performance on `ICL` sequences with held-out query subjects.

Effect of Model Scale. We examine the effect of changing the number of layers $(2, 4, 6, 8)$, the number of heads $(1, 2, 4)$, and the embedding dimension $(128, 256, 512)$ of the model. Fig. 12 compares the final accuracy on `ICL` sequences with held-out subjects when the model is finetuned on `ICL+Grm` sequences with a subset of subjects (same setting as Fig. 4). Overall, we find that i) across these settings, after pretraining on PT sequences, the model succeeds at factual recall but fails at contextual recall, and ii) after FT on ICL+Grm sequences with short, variable grammar length using a subset of subjects, the model succeeds on contextual recall (i.e., good generalization on ICL sequences with held-out query subjects) when model scale is moderate; however, at very small or very large model sizes, this type of FT is not sufficient.

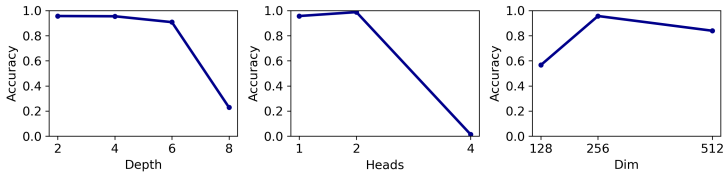

Figure 12: Effect of varying the model depth (left), number of heads (middle), and embedding dimension (right) on the performance on ICL sequences with held-out query subjects, when finetuning the model with ICL+Grm sequences with subset of subjects, keeping other parameters fixed (default: 2 layers, 1 head, 256 dimension). We observe that performance is best at moderate scale, and deteriorates when the scale is much smaller or much larger.

Effect of Increasing the Number of PT Subjects. In Fig. 13, we probe the effect of increasing the number of subjects M used for pretraining. Increasing M improves the performance on ICL sequences with held-out subjects after finetuning. Here, we use grammar sequence length $S = 80$. Consistent with the observations in Fig. 5, we observe that using a sufficiently large S for pretraining leads to comparable ICL performance, across different levels of Markov chain diversity Div.

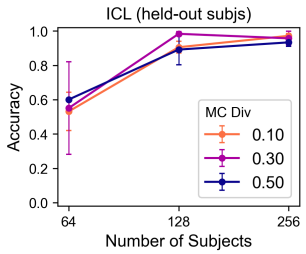

Figure 13: Increasing the number of subjects used while pretraining improves the model's performance on ICL sequences with held-out subjects after finetuning with ICL+Grm sequences.

Effect of Increasing the Number of Attributes. In order to verify that our findings generalize to settings with a higher number of attribute types, we set $L = 32$, while keeping $M = 256$ subjects, and scale the embedding dimension ($256 \to 288$ ($\propto M + L$)) and training duration ($20k \to 40k$) for the larger factual world. We find that our key findings remain consistent in this case. Specifically, the model achieves 100% factual recall accuracy after PT while ICL accuracy is near-zero, i.e., PT alone does not yield contextual recall (Finding 1). After FT on ICL+Grm sequences, ICL accuracy on sequences with held-out query subjects increases to 99.8% (Finding 3).

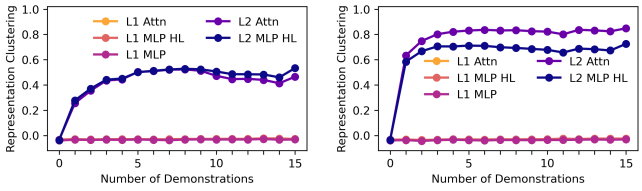

Figure 14: Comparison of clustering strength (see Section 3.3 for details) using representations from different layers of the finetuned 2-layer 1-head transformer with ICL sequences as inputs, as the number of demonstrations is increased (same setting as Fig. 6) with Div ≈ 0.2 (left) and Div ≈ 0.5 (right). We find that layer-2 attention representations cluster most strongly, while layer-1 representations exhibit no clustering based on the attribute type information in the context. Further, using higher Div while pretraining leads to stronger representation clustering after finetuning.

B.3 Details of Representational Analysis

Consider ICL sequences, which are of the form

$$X_\ell = [s_{j_0}, a^\ell_{j_0}, [\text{sep}], s_{j_1}, a^\ell_{j_1}, [\text{sep}], \ldots, s_{j_{N+1}}],$$

for a fixed attribute type ℓ. We sample K such sequences for each $\ell \in [L]$, denoted by X^k_ℓ. Also, define $X_{\ell,t} := [s_{j_0}, a^\ell_{j_0}, [\text{sep}], s_{j_1}, a^\ell_{j_1}, [\text{sep}], \ldots, s_{j_{t+1}}]$, where $t \in [N]$ denotes the number of demonstrations.

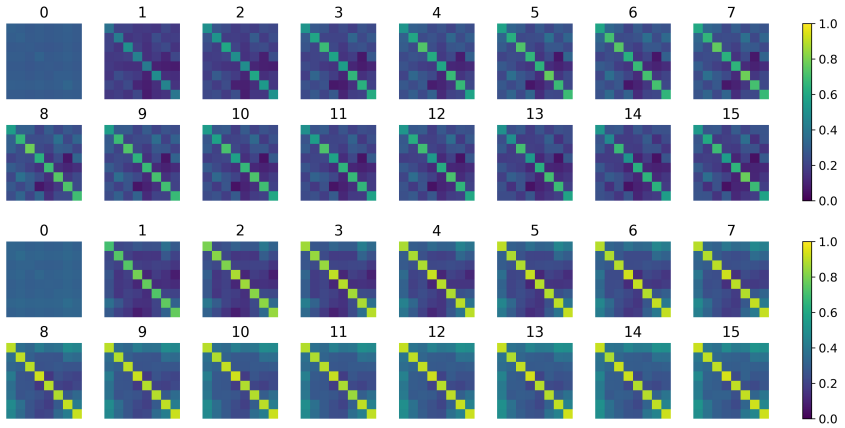

Figure 15: Each subfigure visualizes the cosine similarity for inter- and intra-task representations (from layer-2 attention layer of the finetuned model), $\bar{C}^t(\ell, \ell')$ (see Section 3.3 for details) across attribute types $\ell, \ell' \in [L]$, averaged over 50 sequences for each attribute type (same setting as Fig. 6; top: $\texttt{Div} \approx 0.2$, bottom: $\texttt{Div} \approx 0.5$). We find that as the number of demonstrations is increased (from 0 to 15), the representations of ICL sequences with the same attribute type get clustered together, and the clustering is stronger for higher Div (*i.e.*, more separated attribute-specific Markov shains).

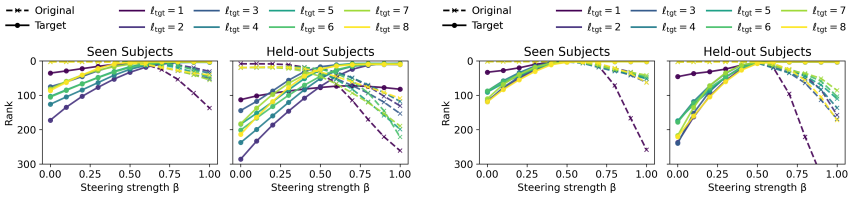

Figure 16: Comparison of rank of the original and target attribute type when steering the model output using the averaged layer-2 attention representations of the finetuned 2-layer 1-head transformer with ICL sequences as inputs at the last token position (see Section 3.3 for details) with $\texttt{Div} \approx 0.2$ (left) and $\texttt{Div} \approx 0.5$ (right), for seen and held-out FT subjects, for each target attribute type. We find that higher Div used while pretraining leads to stronger task vectors and better steering with lower β values.

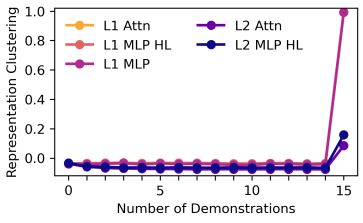

Figure 17: Comparison of clustering strength (using query subject index as the label) using representations from different layers of the finetuned 2-layer 1-head transformer with ICL sequences as inputs, as the number of demonstrations is increased with $\texttt{Div} \approx 0.5$. We find that layer-1 MLP representations cluster most strongly and only at the last position in the context.

Let $f_i(\cdot)$ denote the model's representation at layer i. Consider fixed i, t, and for $\ell, \ell' \in [L]$, define

$$\bar{C}_i^t(\ell, \ell') = \frac{1}{K^2} \sum_{k,k'} \cos\big(f_i(X_{\ell,t}^k),\ f_i(X_{\ell',t}^{k'})\big).$$

Next, the clustering strength is quantified as follows. For fixed i, t, we subsume these and define $v_\ell^k := f_i^t(X_\ell^k)$. Each representation is assigned cluster label based on the attribute type ℓ. Define the

cosine distance $d(v, w) := 1 - \cos(v, w)$. For each point v_ℓ^k, define the intra-cluster dissimilarity

$$a(k, \ell) = \frac{1}{K-1} \sum_{k' \neq k} d(v_\ell^k, v_\ell^{k'}),$$

and the nearest other-cluster dissimilarity

$$b(k, \ell) = \min_{\ell' \neq \ell} \left(\frac{1}{K} \sum_{k'} d(v_\ell^k, v_{\ell'}^{k'}) \right).$$

The silhouette value for a sample is

$$s(k, \ell) = \frac{b(k, \ell) - a(k, \ell)}{\max\{\, a(k, \ell),\, b(k, \ell)\,\}}.$$

Finally, the silhouette score for layer i after t demonstrations are seen is

$$\bar{S}_i^t = \frac{1}{KL} \sum_\ell \sum_k s(k, \ell).$$

C Analysis for the Attention-Only Model

C.1 Constructions for PT and ICL Sequences

Consider PT sequences of the form

$$\boldsymbol{X} = [\boldsymbol{h}(s_{j_\star}), \boldsymbol{h}(g_1), ..., \boldsymbol{h}(r_{\ell_1}), ..., \boldsymbol{h}([\text{sep}]), \boldsymbol{h}(a_{j_\star}^{\ell_1}), \boldsymbol{h}(s_{j_\star}), ..., \boldsymbol{h}([\text{sep}])], \tag{8}$$

where the correct last token prediction is $\boldsymbol{h}(a_{j_\star}^{\ell_{N_{\text{tr}}}})$. The following result shows that there exists an attention-only model that always gives correct predictions at any position.

Proposition 3. *Consider the input $\boldsymbol{X}$ in Eq. (8), and a 1-layer 3-head attention-only model with the weights*

$$\boldsymbol{W}_{KQ}^{rel} = \beta \left(\sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{p} \right) \boldsymbol{h}([sep])^\top, \quad \boldsymbol{W}_{OV}^{rel} = \sum_\ell \sum_j \boldsymbol{h}(u_j^\ell) \boldsymbol{h}(r_\ell)^\top,$$

$$\boldsymbol{W}_{KQ}^{subj} = \beta (\sum_j \boldsymbol{h}(s_j)) \boldsymbol{h}([sep])^\top, \quad \boldsymbol{W}_{OV}^{subj} = -\sum_j \left(\sum_{j' \neq j, \ell} \boldsymbol{h}(a_{j'}^\ell) \right) \boldsymbol{h}(s_j)^\top,$$

$$\begin{aligned} \boldsymbol{W}_{KQ}^{grm} = \beta \Big(& \sum_{\ell, j} \boldsymbol{h}(u_j^\ell) \boldsymbol{h}(u_j^\ell)^\top + \sum_j \boldsymbol{h}(s_j) \boldsymbol{h}(s_j)^\top + \sum_\ell (\boldsymbol{h}(r_\ell) + \boldsymbol{h}([sep]) + \boldsymbol{p}) \boldsymbol{h}(r_\ell)^\top \\ & + \Big(\sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{h}([sep]) + \boldsymbol{p} \Big) \sum_{\tilde{g}} \boldsymbol{h}(\tilde{g})^\top \Big) \end{aligned}$$

$$\begin{aligned} \boldsymbol{W}_{OV}^{grm} = & \sum_j \sum_\ell \boldsymbol{h}(s_j) \boldsymbol{h}(a_j^\ell)^\top + \Big(\sum_{\tilde{g}} \boldsymbol{h}(\tilde{g}) + \sum_\ell \boldsymbol{h}(r_\ell) \Big) \Big(\sum_j \boldsymbol{h}(s_j) + \boldsymbol{h}([sep]) \Big)^\top \\ & + \Big(0.5 \sum_{\tilde{g}} \boldsymbol{h}(\tilde{g}) + \boldsymbol{h}([sep]) \Big) \sum_\ell \boldsymbol{h}(r_\ell)^\top, \end{aligned}$$

where $\boldsymbol{p} := \sum_{i=1}^{S+2} \boldsymbol{h}(p_{-i+1})$. Then, when the scalar $\beta \to \infty$, it always gives the correct prediction for any token position.

Proof. We present a construction for a 3-head model. For simplicity, we present the proof for the last subsequence, *i.e.*, $t \in \{\mathrm{T} - S - 2, \ldots, \mathrm{T}\}$, but it can be easily extended to other cases where $t < \mathrm{T} - S - 2$ as well. Hereafter, we assume that $t \geq \mathrm{T} - S - 2$.

At a high level, we use two heads, which we call the **relation** and **subject** heads, to get the output following $\boldsymbol{h}([\text{sep}])$, *i.e.*, $t = \mathrm{T}$, and the third **grammar** head for other cases $t < \mathrm{T}$. Let $\mathcal{T} := \{\mathrm{T} - S, \ldots, \mathrm{T} - 2\}$.

Consider the **relation** head. Then, when $\beta \to \infty$, the outputs of this head are as follows. When $t = \mathrm{T}$, $g_{\text{rel}}(\boldsymbol{X}) = \boldsymbol{h}(r_{\ell_{N_{\text{tr}}}})$, *i.e.*, the most recent relation token in the sequence, and $f_{\text{rel}}(\boldsymbol{X}) = \sum_{j,\ell_{N_{\text{tr}}}} \boldsymbol{h}(u_j^{\ell_{N_{\text{tr}}}})$, *i.e.*, all attributes of type $\ell_{N_{\text{tr}}}$.

On the other hand, when $t < \mathrm{T}$, $f_{\text{rel}}^t(\boldsymbol{X}) = \frac{1}{t}\sum_{j,\ell_{N_{\text{tr}}}} \boldsymbol{h}(u_j^{\ell_{N_{\text{tr}}}})$.

Consider the **subject** head. Then, when $\beta \to \infty$, the outputs of this head are as follows. When $t = \mathrm{T}$, $g_{\text{subj}}(\boldsymbol{X}) = \boldsymbol{h}(s_{j_\star})$, the subject token in the sequence, and $f_{\text{subj}}(\boldsymbol{X}) = -\Big(\sum_u \boldsymbol{h}(u) - \sum_\ell \boldsymbol{h}(a_{j_\star}^\ell)\Big)$, *i.e.*, negative of all attributes that are not associated with subject $s_{j_\star}$.

On the other hand, when $t < \mathrm{T}$, $f_{\text{subj}}^t(\boldsymbol{X}) = -\frac{1}{t}\Big(\sum_u \boldsymbol{h}(u) - \sum_\ell \boldsymbol{h}(a_{j_\star}^\ell)\Big)$.

Consider the **grammar** head. Let $g_h^t = g_h^t(\boldsymbol{X})$ and similarly $f_h^t := f_h^t(\boldsymbol{X})$. The outputs in this case, when $\beta \to \infty$, are as follows:

- $t = \mathrm{T} - S - 2$: $g_{\text{grm}} = \boldsymbol{h}(a_{j_\star}^{\ell_{N-1}})$, $f_{\text{grm}} = \boldsymbol{h}(s_{j_\star})$
- $t = \mathrm{T} - S - 1$: $g_{\text{grm}} = \boldsymbol{h}(s_{j_\star})$, $f_{\text{grm}} = \sum_{v\in\mathcal{G}\cup\mathcal{R}} \boldsymbol{h}(v)$
- $t \in \mathcal{T}$: $g_{\text{grm}} = \begin{cases} 0.5(\boldsymbol{h}([\text{sep}]) + \boldsymbol{h}(r_\ell)), & \text{if } \boldsymbol{x}_{t-S-2:t} \in \mathcal{R} \\ \boldsymbol{h}([\text{sep}]), & \text{if } \boldsymbol{x}_{t-S-2:t} \notin \mathcal{R} \end{cases}$,
  $f_{\text{grm}} = \begin{cases} \sum_{v\in\mathcal{G}\cup\mathcal{R}} \boldsymbol{h}(v), & \text{if } g_{\text{grm}} = \boldsymbol{h}([\text{sep}]) \\ 0.75\sum_{v\in\mathcal{G}} \boldsymbol{h}(v) + 0.5\sum_{v\in\mathcal{R}} \boldsymbol{h}(v) + 0.5\boldsymbol{h}([\text{sep}]), & \text{if } g_{\text{grm}} = 0.5(\boldsymbol{h}([\text{sep}]) + \boldsymbol{h}(r_{\ell_{N_{\text{tr}}}})) \end{cases}$
- $t = \mathrm{T} - 1$: $g_{\text{grm}} = \boldsymbol{h}(r_{\ell_{N_{\text{tr}}}})$, $f_{\text{grm}} = \boldsymbol{h}([\text{sep}]) + 0.5\sum_{v\in\mathcal{G}} \boldsymbol{h}(\tilde{g})$
- $t = \mathrm{T}$: $g_{\text{grm}} = \frac{1}{\mathrm{T}}\sum_t \boldsymbol{x}_t$, $f_{\text{grm}} = \frac{N_{\text{tr}}-1}{\mathrm{T}}\boldsymbol{h}(s_{j_\star}) + \frac{N_{\text{tr}}}{\mathrm{T}}\Big(1.5\sum_{v\in\mathcal{G}} \boldsymbol{h}(v) + \sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{h}([\text{sep}])\Big)$

Combining the outputs of the individual heads, the final output in different cases is as follows.

$$v^* = \begin{cases} a_{j_\star}^{\ell_{N_{\text{tr}}}}, & \text{if } t = \mathrm{T}, \\ [\text{sep}], & \text{if } t = \mathrm{T}-1, \\ v \sim \text{Unif}(\mathcal{G}\cup\mathcal{R}), & \text{if } t \in \mathcal{T}, \nexists j \in \{t-S-2, ..., t\}, \boldsymbol{x}_j \in \mathcal{R}, \\ v \sim \text{Unif}(\mathcal{G}), & \text{if } t \in \mathcal{T}, \exists j \in \{t-S-2, ..., t\}, \boldsymbol{x}_j \in \mathcal{R}, \\ v \sim \text{Unif}(\mathcal{G}\cup\mathcal{R}), & \text{if } t = \mathrm{T}-S-1, \\ s_{j_\star}, & \text{if } t = \mathrm{T}-S-2. \end{cases}$$

□

Next, consider `ICL` sequences of the form

$$\boldsymbol{X} = [\boldsymbol{h}(s_{j_1}), \boldsymbol{h}([\text{sep}]), \boldsymbol{h}(a_{j_1}^{\ell_\star}), ..., \boldsymbol{h}(s_{j_{N_{\text{ft}}+1}}), \boldsymbol{h}([\text{sep}])], \tag{9}$$

where the correct last token prediction is $\boldsymbol{h}(a_{j_{N_{\text{ft}}+1}}^{\ell_\star})$. The following result shows that there exists an attention-only model that always gives correct predictions for these sequences.

Proposition 4. *Consider the input $\boldsymbol{X}$ in Eq. (9), and a 1-layer 3-head attention-only model with the weights*

$$\boldsymbol{W}_{KQ}^{subj} = \beta\Big(\sum_j \boldsymbol{h}(s_j) + \boldsymbol{h}(p_{-1})\Big)\boldsymbol{h}([sep])^\top, \quad \boldsymbol{W}_{OV}^{subj} = -\sum_j \left(\sum_{j'\neq j,\ell} \boldsymbol{h}(a_{j'}^\ell)\right)\boldsymbol{h}(s_j)^\top,$$

$$\boldsymbol{W}_{KQ}^{rel} = \beta\sum_\ell \sum_{u\in\mathcal{U}_\ell'} \boldsymbol{h}(u)\boldsymbol{h}([sep])^\top, \quad \boldsymbol{W}_{OV}^{rel} = \sum_\ell \Big(\sum_{u\in\mathcal{U}_\ell'} \boldsymbol{h}(u)\Big)\Big(\sum_{u\in\mathcal{U}_\ell'} \boldsymbol{h}(u) + \boldsymbol{h}(r_\ell)\Big)^\top,$$

for the subject and relation heads, and the same weights for the grammar head as in Proposition 3. Here, for each attribute type ℓ, $\mathcal{U}'_\ell := \cup_{j\in\mathcal{S}'}\{a^\ell_j\}$ denotes the set of unique attributes for the subjects in $\mathcal{S}' \subset \mathcal{S}$ seen during finetuning. Then, when the scalar $\beta \to \infty$ and $\mathcal{U}'_\ell = \mathcal{U}$, this model always gives the correct prediction for the last token position.

Proof. We present a construction for a 3-head model, with minimal changes to the construction in the proof of Proposition 1. Specifically, the **grammar** head is unchanged. At a high level, the **subject** head attends to the last/query subject and maps it to the negative of the sum of attributes not associated with it, while the **relation** head attends to the attributes in the context, and maps them to all attributes of the same type $\ell_\star$.

Consider the **subject** head. Using the aforementioned weights, when $\beta \to \infty$, $g_{\text{subj}}(\boldsymbol{X}) = \boldsymbol{h}(s_{j_{N_{\text{ft}}+1}})$, *i.e.*, the query subject, and $f_{\text{subj}}(\boldsymbol{X}) = -\Big(\sum_u \boldsymbol{h}(u) - \sum_\ell \boldsymbol{h}(a^\ell_{j_{N_{\text{ft}}+1}})\Big)$.

Consider the **relation** head. Using the aforementioned weights, when $\beta \to \infty$, $g_{\text{rel}}(\boldsymbol{X}) = \frac{1}{N_{\text{ft}}}\sum_i \boldsymbol{h}(a^{\ell_\star}_{j_i})$, *i.e.*, the average of the attributes that appear in the context, and $f_{\text{rel}}(\boldsymbol{X}) = \sum_{u\in\mathcal{U}_{\ell_\star}} \boldsymbol{h}(u)$.

Combining the outputs of the individual heads, the model output is

$$f(\boldsymbol{X}) = \sum_{u\in\mathcal{U}_{\ell_\star}} \boldsymbol{h}(u) - \Big(\sum_u \boldsymbol{h}(u) - \sum_\ell \boldsymbol{h}(a^\ell_{j_{N_{\text{ft}}+1}})\Big) + \tfrac{1}{\mathrm{T}}\sum_{i\in\mathcal{M}''} \boldsymbol{h}(s_i) + \tfrac{N_{\text{ft}}}{\mathrm{T}}\sum_{v\in\mathcal{G}\cup\mathcal{R}} \boldsymbol{h}(v),$$

where $\mathcal{M}'' \subseteq [M]$. Then, the final prediction $v^* = a^{\ell_\star}_{j_{N_{\text{ft}}+1}}$. □

C.2 Optimization Dynamics for Finetuning

We state the formal version of Theorem 1 below followed by its proof.

Theorem 2. *Consider the two-head attention-only model defined in Eq. (1), initialized with the weights from the construction for* `PT` *sequences in Proposition 1:*

$$\boldsymbol{W}^{subj}_{KQ} = \beta\Big(\sum_j \boldsymbol{h}(s_j)\Big)\boldsymbol{h}([sep])^\top, \qquad \boldsymbol{W}^{subj}_{OV} = -\sum_j\Big(\sum_{j'\neq j,\ell} \boldsymbol{h}(a^\ell_{j'})\Big)\boldsymbol{h}(s_j)^\top,$$

$$\boldsymbol{W}^{rel}_{KQ} = \beta\Big(\sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{p}\Big)\boldsymbol{h}([sep])^\top, \qquad \boldsymbol{W}^{rel}_{OV} = \sum_\ell\Big(\sum_j \boldsymbol{h}(u^\ell_j)\Big)\boldsymbol{h}(r_\ell)^\top.$$

Assume that $\mathcal{U}'_\ell = \mathcal{U}_\ell\ \forall \ell \in [L]$, $M \gg mL, M > M_{ft} \gg U_L$, $\mathrm{T} \leq S+2$. Consider last-token prediction on ICL sequences of the form

$$\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{\mathrm{T}}] = [\boldsymbol{h}(s_{j_1}), \boldsymbol{h}([sep]), \boldsymbol{h}(a^{\ell_\star}_{j_1}), \ldots, \boldsymbol{h}(s_{j_{N_{ft}+1}}), \boldsymbol{h}([sep])], \quad y = a^{\ell_\star}_{j_{N_{ft}+1}}, \tag{10}$$

on the population, where the attribute index and each subject in the context are sampled i.i.d. with replacement from a randomly selected set of M_{ft} subjects. Then, keeping the weight $\boldsymbol{W}^{subj}_{OV}$ fixed, and updating the other weights with one step of GD on $\boldsymbol{W}^{subj}_{KQ}$ with step size η^{subj}_{KQ}, one step of projected GD (right projection with $\boldsymbol{I} - \boldsymbol{h}([sep])\boldsymbol{h}([sep])^\top$) on $\boldsymbol{W}^{rel}_{OV}$ with step size η^{rel}_{OV}, and two steps of GD on $\boldsymbol{W}^{rel}_{KQ}$ with step size η^{rel}_{KQ}, the updated weights satisfy

$$\boldsymbol{W}^{subj}_{KQ} \approx \beta\left(\sum_j \boldsymbol{h}(s_j) + \alpha_1 \boldsymbol{h}(p_{-1})\right)\boldsymbol{h}([sep])^\top,$$

$$\boldsymbol{W}^{rel}_{KQ} \approx \beta\left(\sum_\ell \boldsymbol{h}(r_\ell) + \alpha_3\left(\tfrac{2}{3}\tfrac{1}{LU_L}\sum_u \boldsymbol{h}(u) + \tfrac{1}{m-1}\sum_{i=1}^{m-1} \boldsymbol{p}_{3i+2} - \tfrac{1}{\mathrm{T}}\boldsymbol{p} - \tfrac{1}{3}\tfrac{1}{M}\sum_s \boldsymbol{h}(s)\right)\right)\boldsymbol{h}([sep])^\top,$$

$$\boldsymbol{W}^{rel}_{OV} \approx \sum_\ell\Big(\sum_j \boldsymbol{h}(u^\ell_j)\Big)\boldsymbol{h}(r_\ell)^\top + \alpha_2\Big((m-1)\sum_{\ell=1}^{L}\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u^\ell_j)\Big)\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u^\ell_j)\Big)^\top$$
$$+ \tfrac{U_L^2}{M_{ft}}\sum_{j=1}^{M_{ft}}\Big(\sum_{\ell=1}^{L}\boldsymbol{h}(a^\ell_j)\Big)\boldsymbol{h}(s_j)^\top\Big),$$

where $\boldsymbol{p}_i = \boldsymbol{h}(p_{i-\mathrm{T}+1})$, $\alpha_1 := \eta_{KQ}^{subj} m^{-1}\left(1 - U_L^{-1}\right)$, $\alpha_2 = \eta_{OV}^{rel}(LTU_L^2)^{-1}$, *and* $\alpha_3 := \eta_{KQ}^{rel}(m-1)^2\alpha_2$. *Further, when* $\beta \to \infty$, *this model gets perfect accuracy on* `ICL` *sequences with seen or held-out query subjects.*

Proof. Note that for a given `ICL` sequence $\boldsymbol{X}$, the query subject $s_{j_{N_{\mathrm{ft}}+1}}$ appears at the fixed relative position. The last-token negative log-likelihood is

$$\mathcal{L}(\boldsymbol{X}, y) = -\boldsymbol{h}(y)^\top f(\boldsymbol{X}) + \log \sum_{v \in \mathcal{V}} \exp\left(\boldsymbol{h}(v)^\top f(\boldsymbol{X})\right),$$

with gradient

$$\boldsymbol{u} := \nabla_f \mathcal{L} = \sum_{v \in \mathcal{V}} \pi_v \, \boldsymbol{h}(v) - \boldsymbol{h}(y), \qquad \pi_v = \frac{e^{\boldsymbol{h}(v)^\top f}}{\sum_u e^{\boldsymbol{h}(u)^\top f}}.$$

Recall that $f(\boldsymbol{X}) = \sum_{k=1}^{H} \boldsymbol{W}_{OV}^k g_k(\boldsymbol{X})$, where $g_k(\boldsymbol{X}) = \sum_{i=1}^{\mathrm{T}} \alpha_i^k \boldsymbol{x}_i$, $\alpha_i^k := \boldsymbol{\varphi}(\lambda_i^k)$, $\lambda_i^k := (\boldsymbol{x}_i + \boldsymbol{p}_i)^\top \boldsymbol{W}_{KQ}^k \boldsymbol{h}([\mathrm{sep}])$.

Next, we compute the gradients with respect to $\boldsymbol{W}_{OV}^k$ and $\boldsymbol{W}_{KQ}^k$.

We have that

$$\nabla_{\boldsymbol{W}_{OV}^k} \mathcal{L} = \boldsymbol{u} g_k(\boldsymbol{X})^\top = \boldsymbol{u} \sum_{i=1}^{\mathrm{T}} \alpha_i^k \boldsymbol{x}_i^\top.$$

Further, for each i, k,

$$\frac{\partial \mathcal{L}}{\partial \alpha_i^k} = \left\langle \nabla_{g_k} \mathcal{L}, \frac{\partial g_k}{\partial \alpha_i^k} \right\rangle = \boldsymbol{u}^\top \boldsymbol{W}_{OV}^k \boldsymbol{x}_i =: r_i^k,$$

where we use $\nabla_{g_k} \mathcal{L} = (\boldsymbol{W}_{OV}^k)^\top \nabla_f \mathcal{L} = (\boldsymbol{W}_{OV}^k)^\top \boldsymbol{u}$. Next, we have that

$$\frac{\partial \alpha_i^k}{\partial \lambda_j^k} = \alpha_i^k (\mathbb{1}[i = j] - \alpha_j^k).$$

Therefore,

$$\frac{\partial \mathcal{L}}{\partial \lambda_j^k} = \sum_{i=1}^{\mathrm{T}} \frac{\partial \mathcal{L}}{\partial \alpha_i^k} \frac{\partial \alpha_i^k}{\partial \lambda_j^k} = \sum_{i=1}^{\mathrm{T}} r_i^k \alpha_i^k (\mathbb{1}[i = j] - \alpha_j^k) = r_j^k \alpha_j^k - \alpha_j^k \sum_{i=1}^{\mathrm{T}} \alpha_i^k r_i^k = \alpha_j^k \left(r_j^k - \bar{r}^k\right),$$

where we define $\bar{r}^k := \sum_{i=1}^{\mathrm{T}} \alpha_i^k r_i^k$. Finally, we have that

$$\nabla_{\boldsymbol{W}_{KQ}^k} \mathcal{L} = \sum_{j=1}^{\mathrm{T}} \frac{\partial \mathcal{L}}{\partial \lambda_j^k} \nabla_{\boldsymbol{W}_{KQ}^k} \lambda_j^k = \beta \sum_{j=1}^{\mathrm{T}} \alpha_j^k (r_j^k - \bar{r}^k) (\boldsymbol{x}_j + \boldsymbol{p}_j) \boldsymbol{h}([\mathrm{sep}])^\top. \tag{11}$$

We next analyze the gradient updates for two iterations of finetuning. We keep $\boldsymbol{W}_{OV}^{\mathrm{subj}}$ fixed throughout.

First Iteration. For the first iteration, we analyze the updates for $\boldsymbol{W}_{KQ}^{\mathrm{subj}}$, $\boldsymbol{W}_{KQ}^{\mathrm{rel}}$, and $\boldsymbol{W}_{OV}^{\mathrm{rel}}$.

Subject Head ($\boldsymbol{W}_{KQ}^{\mathrm{subj}}$). We first focus on the *subject head*. Recall from the construction for `PT` sequences that $\boldsymbol{W}_{KQ}^{\mathrm{subj}} = \beta\Big(\sum_{j \in \mathcal{S}} \boldsymbol{h}(s_j)\Big)\boldsymbol{h}([\mathrm{sep}])^\top$. Let $\mathcal{I}_s := \{i : \exists s \in \mathcal{S}, \boldsymbol{x}_i = \boldsymbol{h}(s)\}$ denote the set of subject positions, with $|\mathcal{I}_s| = N_{\mathrm{ft}} + 1 =: m$. For $i \in \mathcal{I}_s$, let $J(i)$ denote the subject index j_i in the context, such that $j_i \in [M]$, $s_{j_i} \in \mathcal{S}$. For $\beta \to \infty$, the resulting attention weights are

$$\alpha_i = \begin{cases} \frac{1}{m}, & i \in \mathcal{I}_s, \\ 0, & i \notin \mathcal{I}_s, \end{cases}$$

where we omit the superscript k for brevity. Substituting into Eq. (11), we obtain

$$\nabla_{\boldsymbol{W}_{KQ}^{\mathrm{subj}}} \mathcal{L} = \frac{\beta}{m} \sum_{i \in \mathcal{I}_s} (r_i - \bar{r}) (\boldsymbol{x}_i + \boldsymbol{p}_i) \boldsymbol{h}([\mathrm{sep}])^\top, \qquad \bar{r} = \frac{1}{m} \sum_{i \in \mathcal{I}_s} r_i.$$

Let $\hat{r}_i := r_i - \sum_{v\in\mathcal{V}} \pi_v \boldsymbol{h}(v)^\top \boldsymbol{W}_{OV}\boldsymbol{x}_i$. Let $\mathcal{A}_j$ denote the set of all attributes associated with subject s_j. We have that

$$\sum_{v\in\mathcal{V}} \pi_v \boldsymbol{h}(v)^\top \boldsymbol{W}_{OV}^{\text{subj}} \boldsymbol{x}_i = \mathbb{1}\left[i \in \mathcal{I}_s\right] \sum_{u\in\mathcal{A}_{J(i)}} \pi_u.$$

From Lemma C.1, $\pi_u \le \frac{1}{V+LU_L(C\exp^{-1}-1)}$. Then, we have

$$|\hat{r}_i - r_i| \le \frac{L}{V + LU_L(C\exp^{-1}-1)} \ll \frac{1}{m},$$

since $V \ge M + U_L L \gg mL$. We also define

$$\nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \hat{\mathcal{L}} := \frac{\beta}{m} \sum_{i\in\mathcal{I}_s} (\hat{r}_i - \bar{\hat{r}})(\boldsymbol{x}_i + \boldsymbol{p}_i)\, \boldsymbol{h}([\text{sep}])^\top, \qquad \bar{\hat{r}} = \frac{1}{m}\sum_{i\in\mathcal{I}_s} \hat{r}_i.$$

Then, we have that

$$\begin{aligned}\frac{1}{\beta}\|\nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \hat{\mathcal{L}} - \nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \mathcal{L}\| &\le \frac{1}{m}\left(\|\sum_i (r_i - \hat{r}_i)(\boldsymbol{x}_i + \boldsymbol{p}_i)\boldsymbol{h}([\text{sep}])^\top\| + \|(\bar{r} - \bar{\hat{r}})\sum_i (\boldsymbol{x}_i + \boldsymbol{p}_i)\boldsymbol{h}([\text{sep}])^\top\|\right) \\ &\le \frac{\sqrt{2}}{m}\left(m\max_i |r_i - \hat{r}_i| + m\max_i |r_i - \hat{r}_i|\right) \ll \frac{2\sqrt{2}}{m}.\end{aligned}$$

Therefore, in the next part, we analyze $\nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \hat{\mathcal{L}}$ (as we will see, $\beta^{-1}\|\nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \hat{\mathcal{L}}\|$ is $\Theta(1/m)$, *i.e.* it dominates the norm of the difference). We have that

$$\hat{r}_i = -\begin{cases} \mathbb{1}\left[a_{J(i)}^{\ell_\star} = y\right], & i \in \mathcal{I}_s, \\ 0, & \text{o.w.}\end{cases}$$

Let $q = 3N_{\text{ft}}$ denote the index of the query subject. Then, we have that

$$\hat{r}_i - \bar{\hat{r}} = -\begin{cases} 1 - \frac{1}{m}\sum_{k\in\mathcal{I}_s} \mathbb{1}\left[a_{J(k)}^{\ell_\star} = y\right], & i = q, \\ \mathbb{1}\left[a_{J(i)}^{\ell_\star} = y\right] - \frac{1}{m}\sum_{k\in\mathcal{I}_s} \mathbb{1}\left[a_{J(k)}^{\ell_\star} = y\right], & i \in \mathcal{I}_s \setminus \{q\}, \\ 0, & \text{o.w.}\end{cases}$$

Let $C_y := \sum_{i=1}^m \mathbb{1}\left[a_{j_i}^{\ell_\star} = y\right]$. Then,

$$\begin{aligned}\nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \hat{\mathcal{L}} = -\frac{\beta}{m}\Big(&\sum_{k=1}^{m}(\mathbb{1}\left[a_{j_k}^{\ell_\star} = y\right] - m^{-1}C_y)\boldsymbol{h}(s_{j_k}) + \sum_{k=1}^{m-1}(\mathbb{1}\left[a_{j_k}^{\ell_\star} = y\right] - m^{-1}C_y)\boldsymbol{p}_{3k} \\ &+ (1 - m^{-1}C_y)\boldsymbol{h}(p_{-1})\Big)\boldsymbol{h}([\text{sep}])^\top. \end{aligned} \tag{12}$$

We first calculate some expectations. Under i.i.d. context sampling with replacement, conditioned on y, we have $\Pr(a_i^{\ell_\star} = y \mid y) = \frac{1}{U_L}$, and hence $\mathbb{E}[C_y \mid y] = \frac{m}{U_L}$. Further, averaging over y gives $\mathbb{E}[C_y] = \frac{m}{U_L}$.

Let $\mu_s(y) := \mathbb{E}[\boldsymbol{h}(s) \mid a^{\ell_\star} = y]$. Under i.i.d. context sampling with replacement,

$$\mathbb{E}\left[\sum_i \mathbb{1}\left[a_i^{\ell_\star} = y\right]\boldsymbol{h}(s_i)\middle| y\right] = \sum_i \mathbb{E}[\mathbb{1}\left[a_i^{\ell_\star} = y\right] s_i \mid y] = m\frac{1}{U_L}\mu_s(y),$$

since $\Pr(a_i^{\ell_\star} = y \mid y) = 1/U_L$. Hence,

$$\mathbb{E}\left[\sum_i \mathbb{1}\left[a_i^{\ell_\star} = y\right]\boldsymbol{h}(s_i)\right] = \frac{m}{U_L}\mathbb{E}[\mu_s(y)].$$

Since each value appears on M/U_L subjects, $\mathbb{E}[\mu_s(y)] = \frac{1}{M}\sum_{j=1}^{M} \boldsymbol{h}(s_j)$, and therefore

$$\mathbb{E}\left[\sum_i \mathbb{1}\left[a_i^{\ell_\star} = y\right] \boldsymbol{h}(s_i)\right] = \frac{m}{U_L M}\sum_{j=1}^{M} \boldsymbol{h}(s_j).$$

Taking expectation in Eq. (12) over the context, and substituting these to simplify, we have that

$$\mathbb{E}\big[\nabla_{\boldsymbol{W}_{KQ}^{\text{subj}}} \hat{\mathcal{L}}\big] = -\frac{\beta}{m}\Big(\Big(\frac{m}{U_L M} - \frac{1}{U_L M}m\Big)\sum_j \boldsymbol{h}(s_j) + \sum_{k=1}^{m-1}\Big(\frac{1}{U_L} - \frac{1}{U_L}\Big)\boldsymbol{p}_{3k}$$
$$+ (1 - U_L^{-1})\boldsymbol{h}(p_{-1})\Big)\boldsymbol{h}([\text{sep}])^\top = -\frac{\beta}{m}\left(1 - \frac{1}{U_L}\right)\boldsymbol{h}(p_{-1})\boldsymbol{h}([\text{sep}])^\top.$$

Since $\alpha_1 = \eta_{KQ}^{\text{subj}} m^{-1}\left(1 - U_L^{-1}\right)$, the updated weight matrix is

$$\boldsymbol{W}_{KQ}^{\text{subj}} \approx \beta\left(\sum_j \boldsymbol{h}(s_j) + \alpha_1 \boldsymbol{h}(p_{-1})\right)\boldsymbol{h}([\text{sep}])^\top.$$

Relation Head. Next, we consider the *relation* head. Recall from the construction for PT sequences that $\boldsymbol{W}_{KQ}^{\text{rel}} = \beta\left(\sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{p}\right)\boldsymbol{h}([\text{sep}])^\top$. Since $S + 2 \geq \mathrm{T}$, first consider the update for the KQ part.

We have that

$$r_i = \frac{1}{\mathrm{T}}\sum_i \boldsymbol{u}^\top\Big(\sum_\ell \boldsymbol{h}(u_\ell)\Big)\mathbb{1}\left[\boldsymbol{x}_i = \boldsymbol{h}(r_\ell)\right] = 0,$$

since the context does not contain a relation token. This means that the gradient is zero and therefore, $\boldsymbol{W}_{KQ}^{\text{rel}}$ is not updated in this iteration.

$\boldsymbol{W}_{OV}^{\text{rel}}$. Next, consider the update for the OV part. We have that

$$\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\mathcal{L} = \frac{1}{\mathrm{T}}\sum_{i=1}^{\mathrm{T}}\Big(\sum_{v\in\mathcal{V}}\pi_v \boldsymbol{h}(v) - \boldsymbol{h}(y)\Big)\boldsymbol{x}_i^\top.$$

Define

$$\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\hat{\mathcal{L}} = -\frac{1}{\mathrm{T}}\boldsymbol{h}(y)\left(\sum_i \boldsymbol{h}(a_i^{\ell_\star}) + \sum_i \boldsymbol{h}(s_i) + m\boldsymbol{h}([\text{sep}])\right)^\top.$$

Then, using Lemma C.1, we have that

$$\|\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\hat{\mathcal{L}} - \nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\mathcal{L}\| \leq \sum_v \pi_v \leq \frac{V}{V + (Ce^{-1} - 1)LU_L} \ll 1,$$

since $C \gg e$. Therefore, we analyze $\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\hat{\mathcal{L}}$ hereon (as we will see, $\|\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\hat{\mathcal{L}}\| = \Theta(\sqrt{U_L L}) > 1$, *i.e.* it dominates the norm of the difference). We have that

$$\mathbb{E}[\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\hat{\mathcal{L}}] = -\frac{m-1}{LTU_L^2}\sum_{\ell=1}^{L}\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)^\top - \frac{m-1}{LTU_L M_{\text{ft}}}\sum_{\ell=1}^{L}\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)\Big(\sum_{j=1}^{M_{\text{ft}}}\boldsymbol{h}(s_j)\Big)^\top$$
$$- \frac{1}{LTM_{\text{ft}}}\sum_{j=1}^{M_{\text{ft}}}\Big(\sum_{\ell=1}^{L}\boldsymbol{h}(a_j^\ell)\Big)\boldsymbol{h}(s_j)^\top - \frac{m}{LTU_L}\sum_{\ell=1}^{L}\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\boldsymbol{h}([\text{sep}])^\top.$$

We right project this gradient onto the subspace orthogonal to $\boldsymbol{h}([\text{sep}])$, and since $M_{\text{ft}} \gg U_L$, $\lambda := (LTU_L^2)^{-1}$, we have

$$-\mathbb{E}[\nabla_{\boldsymbol{W}_{OV}^{\text{rel}}}\hat{\mathcal{L}}](\boldsymbol{I} - \boldsymbol{h}([\text{sep}])\boldsymbol{h}([\text{sep}])^\top) \approx \lambda\Big((m-1)\sum_{\ell=1}^{L}\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)^\top$$
$$+ \frac{U_L^2}{M_{\text{ft}}}\sum_{j=1}^{M_{\text{ft}}}\Big(\sum_{\ell=1}^{L}\boldsymbol{h}(a_j^\ell)\Big)\boldsymbol{h}(s_j)^\top\Big).$$

Since $\alpha_2 = \eta_{OV}^{\text{rel}}(LTU_L^2)^{-1}$, the updated weight matrix is

$$\boldsymbol{W}_{OV}^{\text{rel}} \approx \sum_{\ell}\Big(\sum_{j}\boldsymbol{h}(u_j^\ell)\Big)\boldsymbol{h}(r_\ell)^\top + \alpha_2\Big((m-1)\sum_{\ell=1}^{L}\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)\Big(\sum_{j=1}^{U_L}\boldsymbol{h}(u_j^\ell)\Big)^\top + \frac{U_L^2}{M_{\text{ft}}}\sum_{j=1}^{M_{\text{ft}}}\Big(\sum_{\ell=1}^{L}\boldsymbol{h}(a_j^\ell)\Big)\boldsymbol{h}(s_j)^\top\Big).$$

Second Iteration. In this case, we only update $\boldsymbol{W}_{KQ}^{\text{rel}}$, keeping the other weights fixed.

$\boldsymbol{W}_{KQ}^{\text{rel}}$. We use the updated $\boldsymbol{W}_{OV}^{\text{rel}}$ to compute the gradient wrt $\boldsymbol{W}_{KQ}^{\text{rel}}$. Since $\boldsymbol{W}_{KQ}^{\text{rel}}$ remains unchanged in the first iteration, $\alpha_i^{\text{rel}} = \frac{1}{\mathrm{T}}$, and we have

$$\nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\mathcal{L} = \tfrac{\beta}{\mathrm{T}}\sum_{i=1}^{\mathrm{T}}(r_i-\bar{r})\,(\boldsymbol{x}_i+\boldsymbol{p}_i)\,\boldsymbol{h}([\text{sep}])^\top, \quad r_i := \Big(\sum_{v\in\mathcal{V}}\pi_v\,\boldsymbol{h}(v)-\boldsymbol{h}(y)\Big)^\top\boldsymbol{W}_{OV}^{\text{rel}}\boldsymbol{x}_i, \quad \bar{r} := \tfrac{1}{\mathrm{T}}\sum_i r_i.$$

Also define

$$\nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\hat{\mathcal{L}} = \frac{\beta}{\mathrm{T}}\sum_{i=1}^{\mathrm{T}}(\hat{r}_i-\bar{\hat{r}})(\boldsymbol{x}_i+\boldsymbol{p}_i)\boldsymbol{h}([\text{sep}])^\top, \qquad \hat{r}_i := -\boldsymbol{h}(y)^\top\boldsymbol{W}_{OV}^{\text{rel}}\boldsymbol{x}_i, \qquad \bar{\hat{r}} := \frac{1}{\mathrm{T}}\sum_i\hat{r}_i.$$

Following similar steps as in the update for the subject head, using Lemma C.2, we have that

$$(\beta\alpha_2)^{-1}\|\nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\hat{\mathcal{L}} - \nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\mathcal{L}\| \le 2\sqrt{2}\alpha_2^{-1}\max_i|r_i-\hat{r}_i| \le (\max_{u\in\mathcal{U}}\pi_u)\left((m-1)^2U_L + mL\frac{U_L^2}{M_{\text{ft}}}\right) \le \frac{em^2U_L}{M} \ll m^2,$$

since $M > M_{\text{ft}} \gg U_L$. Therefore, we analyze $\nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\hat{\mathcal{L}}$ hereafter (as we will see, $(\beta\alpha_2)^{-1}\|\nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\hat{\mathcal{L}}\| = \Theta(m^2)$, *i.e.* it dominates the norm of the difference). We have that

$$\hat{r}_i = -\alpha_2\begin{cases} m-1, & \boldsymbol{x}_i\in\mathcal{U}_{\ell_\star} \\ \frac{U_L^2}{M_{\text{ft}}}\mathbb{1}\left[a_i^{\ell_\star}=y\right], & \boldsymbol{x}_i\in\mathcal{S}\end{cases}, \qquad \bar{\hat{r}} = -\frac{\alpha_2}{\mathrm{T}}\left((m-1)^2 + \frac{U_L^2}{M_{\text{ft}}}\sum_i\mathbb{1}\left[a_i^{\ell_\star}=y\right]\right).$$

Therefore,

$$\hat{r}_i - \bar{\hat{r}} = -\alpha_2\begin{cases}(m-1)\left(1-\frac{m-1}{\mathrm{T}}\right) - \frac{U_L^2}{\mathrm{T}M_{\text{ft}}}\sum_i\mathbb{1}\left[a_i^{\ell_\star}=y\right], & \boldsymbol{x}_i\in\mathcal{U}_{\ell_\star}, \\ \frac{U_L^2}{M_{\text{ft}}}\mathbb{1}\left[a_i^{\ell_\star}=y\right] - \frac{U_L^2}{\mathrm{T}M_{\text{ft}}}\sum_i\mathbb{1}\left[a_i^{\ell_\star}=y\right] - \frac{(m-1)^2}{\mathrm{T}}, & \boldsymbol{x}_i\in\mathcal{S}.\end{cases}$$

Let $\lambda_m := (m-1)\left(1-\frac{m-1}{\mathrm{T}}\right)$. Then, $\nabla_{\boldsymbol{W}_{KQ}^{\text{rel}}}\hat{\mathcal{L}}$ can be written as

$$-\frac{\beta\alpha_2}{\mathrm{T}}\Big(\sum_{i=1}^{m-1}\left(\lambda_m - \tfrac{U_L^2}{\mathrm{T}M_{\text{ft}}}C_y\right)(\boldsymbol{h}(a_{j_i}^{\ell_\star})+\boldsymbol{p}_{3i+2}) + \sum_{i=1}^{m}\left(\tfrac{U_L^2}{M_{\text{ft}}}\mathbb{1}\left[a_i^{\ell_\star}=y\right] - \tfrac{U_L^2}{\mathrm{T}M_{\text{ft}}}C_y - \tfrac{(m-1)^2}{\mathrm{T}}\right)(\boldsymbol{h}(s_{j_i})+\boldsymbol{p}_{3i})\Big)\boldsymbol{h}([\text{sep}])^\top.$$

We have that

$$\mathbb{E}[C_y\boldsymbol{h}(y)] = \mathbb{E}[\mathbb{E}[C_y\boldsymbol{h}(y)\mid y]] = \mathbb{E}[\boldsymbol{h}(y)\mathbb{E}[C_y\mid y]] = \frac{m}{U_L}\mathbb{E}[\boldsymbol{h}(y)] = \frac{m}{LU_L^2}\sum_u\boldsymbol{h}(u).$$

The expectations of the terms with attribute, subject and positional tokens can be simplified as follows.

First, for the attribute term, we have

$$\mathbb{E}\left[\sum_{i=1}^{m-1}\left(\lambda_m - \tfrac{U_L^2}{\mathrm{T}M_{\mathrm{ft}}}C_y\right)\boldsymbol{h}(a_{j_i}^{\ell_\star})\right] = \lambda_m(m-1)\tfrac{1}{LU_L}\sum_u \boldsymbol{h}(u) - \tfrac{U_L^2}{\mathrm{T}M_{\mathrm{ft}}}(m-1)\mathbb{E}[C_y\boldsymbol{h}(y)]$$
$$= \left((m-1)\left(1-\tfrac{m-1}{\mathrm{T}}\right) - \tfrac{mU_L}{\mathrm{T}M_{\mathrm{ft}}}\right)\frac{m-1}{LU_L}\sum_u \boldsymbol{h}(u),$$
$$\approx \frac{2(m-1)^2}{3}\frac{1}{LU_L}\sum_u \boldsymbol{h}(u),$$

where the last step follows as $\mathrm{T} = 3m-1$ and $M_{\mathrm{ft}} \gg U_L$.

Next, for the subject term, we have

$$\mathbb{E}\left[\sum_{i=1}^{m}\left(\tfrac{U_L^2}{M_{\mathrm{ft}}}\mathbb{1}\left[a_i^{\ell_\star} = y\right] - \tfrac{U_L^2}{\mathrm{T}M_{\mathrm{ft}}}C_y - \tfrac{(m-1)^2}{\mathrm{T}}\right)\boldsymbol{h}(s_{j_i})\right] = \tfrac{U_L^2}{M_{\mathrm{ft}}}\mathbb{E}\sum_{i=1}^{m}\mathbb{1}\left[a_i^{\ell_\star} = y\right]\boldsymbol{h}(s_{j_i})$$
$$- \mathbb{E}\sum_{i=1}^{m}\left(\tfrac{U_L^2}{\mathrm{T}M_{\mathrm{ft}}}C_y + \tfrac{(m-1)^2}{\mathrm{T}}\right)\boldsymbol{h}(s_{j_i}) = \left(\tfrac{U_L}{M_{\mathrm{ft}}} - \tfrac{mU_L}{\mathrm{T}M_{\mathrm{ft}}} - \tfrac{(m-1)^2}{\mathrm{T}}\right)\tfrac{m}{M}\sum_s \boldsymbol{h}(s)$$
$$\approx -\tfrac{(m-1)^2}{3}\tfrac{1}{M}\sum_s \boldsymbol{h}(s).$$

Next, for the positional encoding term, we have

$$\mathbb{E}\left[\sum_{i=1}^{m-1}(m-1)\boldsymbol{p}_{3i+2} + \tfrac{U_L^2}{M_{\mathrm{ft}}}\sum_{i=1}^{m}\mathbb{1}\left[a_i^{\ell_\star} = y\right]\boldsymbol{p}_{3i} - \sum_{i=1}^{\mathrm{T}}\left(\tfrac{U_L^2}{\mathrm{T}M_{\mathrm{ft}}}C_y + \tfrac{(m-1)^2}{\mathrm{T}}\right)\boldsymbol{p}_i\right]$$
$$= (m-1)\sum_{i=1}^{m-1}\boldsymbol{p}_{3i+2} + \tfrac{mU_L}{M_{\mathrm{ft}}}\sum_{i=1}^{m}\boldsymbol{p}_{3i} - \tfrac{1}{\mathrm{T}}\left(\tfrac{mU_L}{M_{\mathrm{ft}}} + (m-1)^2\right)\sum_i \boldsymbol{p}_i$$
$$= \left(m-1-\tfrac{mU_L}{M_{\mathrm{ft}}}\right)\sum_{i=1}^{m-1}\boldsymbol{p}_{3i+2} + \left(\tfrac{mU_L}{M_{\mathrm{ft}}}\left(1-\mathrm{T}^{-1}\right) - \tfrac{(m-1)^2}{\mathrm{T}}\right)\boldsymbol{p}$$
$$\approx (m-1)\sum_{i=1}^{m-1}\boldsymbol{p}_{3i+2} - \frac{(m-1)^2}{\mathrm{T}}\boldsymbol{p}.$$

Combining these terms, we have that

$$\mathbb{E}[\nabla_{\boldsymbol{W}_{KQ}^{\mathrm{rel}}}\hat{\mathcal{L}}] \approx -(m-1)^2\alpha_2\beta\Big(\tfrac{2}{3}\tfrac{1}{LU_L}\sum_u \boldsymbol{h}(u) + \tfrac{1}{m-1}\sum_{i=1}^{m-1}\boldsymbol{p}_{3i+2} - \tfrac{1}{\mathrm{T}}\boldsymbol{p} - \tfrac{1}{3}\tfrac{1}{M}\sum_s \boldsymbol{h}(s)\Big)\boldsymbol{h}([\mathrm{sep}])^\top.$$

Since $\alpha_3 = \eta_{KQ}^{\mathrm{rel}}(m-1)^2\alpha_2$, the updated weight matrix is

$$\boldsymbol{W}_{KQ}^{\mathrm{rel}} \approx \beta\Big(\sum_\ell \boldsymbol{h}(r_\ell) + \alpha_3\Big(\tfrac{2}{3}\tfrac{1}{LU_L}\sum_u \boldsymbol{h}(u) + \tfrac{1}{m-1}\sum_{i=1}^{m-1}\boldsymbol{p}_{3i+2} - \tfrac{1}{\mathrm{T}}\boldsymbol{p} - \tfrac{1}{3}\tfrac{1}{M}\sum_s \boldsymbol{h}(s)\Big)\Big)\boldsymbol{h}([\mathrm{sep}])^\top.$$

Using these updated weights, following similar steps as in the proof of Proposition 4, we can show that when $\beta \to \infty$, the model attains perfect accuracy on ICL sequences with seen or held-out subjects. □

C.2.1 Helper Lemmas

Lemma C.1. *Under the same conditions as Theorem 1, assuming $M \geq CU_L$, where $C > e$ is a constant, and using the weights*

$$\boldsymbol{W}_{KQ}^{subj} = \beta\Big(\sum_j \boldsymbol{h}(s_j)\Big)\boldsymbol{h}([sep])^\top, \qquad \boldsymbol{W}_{OV}^{subj} = -\sum_j\Big(\sum_{j'\neq j,\ell}\boldsymbol{h}(a_{j'}^{\ell})\Big)\boldsymbol{h}(s_j)^\top,$$
$$\boldsymbol{W}_{KQ}^{rel} = \beta\Big(\sum_\ell \boldsymbol{h}(r_\ell) + \boldsymbol{p}\Big)\boldsymbol{h}([sep])^\top, \qquad \boldsymbol{W}_{OV}^{rel} = \sum_\ell\Big(\sum_j \boldsymbol{h}(u_j^{\ell})\Big)\boldsymbol{h}(r_\ell)^\top,$$

the prediction probabilities π_v are bounded as:

$$\pi_v \le \frac{1}{V + (Ce^{-1} - 1)LU_L}.$$

Proof. Let $m := N_{\text{ft}} + 1$ denote the number of subject positions in the `ICL` sequence, and let k_j be the multiplicity of subject j among these.

When $\beta \to \infty$, the subject head attends uniformly over the m subject positions, hence

$$g_{\text{subj}}(\boldsymbol{X}) = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{h}(s_{j_i}).$$

Therefore

$$f_{\text{subj}}(\boldsymbol{X}) = \boldsymbol{W}_{OV}^{\text{subj}} g_{\text{subj}}(\boldsymbol{X}) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j'\neq j_i,\ell} \boldsymbol{h}(a_{j'}^{\ell}).$$

Moreover, since $\boldsymbol{h}(v)^\top \boldsymbol{W}_{OV}^{\text{rel}} = 0$ for all attributes other than the relation tokens, and since relation tokens don't appear in the `ICL` sequence, the relation head does not contribute to the logits. Therefore,

$$\boldsymbol{h}(a_j^{\ell})^\top f(\boldsymbol{X}) = \boldsymbol{h}(a_j^{\ell})^\top f_{\text{subj}}(\boldsymbol{X}) = -\frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\left[j \neq j_i\right] = -\Big(1 - \frac{k_j}{m}\Big) =: -c_j. \tag{13}$$

Also, note that for any $v \notin \mathcal{U}$, the logit is 0. Next, let $Z := \sum_{u\in\mathcal{V}} \exp(\boldsymbol{h}(u)^\top f(\boldsymbol{X}))$. Using Eq. (13), we have

$$\begin{aligned} Z &= \sum_j \sum_{\ell=1}^{L} \exp(\boldsymbol{h}(a_j^{\ell})^\top f(\boldsymbol{X})) + (V - LU_L) = \sum_j \sum_{\ell=1}^{L} e^{-c_j} + (V - LU_L) \\ &= L\sum_j e^{-c_j} + (V - LU_L) = Le^{-1}\sum_j e^{k_j/m} + (V - LU_L). \end{aligned}$$

Hence, for any attribute token a_j^{ℓ},

$$\pi_{a_j^{\ell}} = \frac{\exp(\boldsymbol{h}(a_j^{\ell})^\top f(\boldsymbol{X}))}{Z} \le \frac{e^{-c_j}}{Le^{-1}\sum_{j'} e^{k_{j'}/m} + (V - LU_L)} = \frac{e^{k_j/m}}{L\sum_{j'} e^{k_{j'}/m} + e(V - LU_L)}.$$

Since $\sum_{j'} e^{k_{j'}/m} \ge M$, and $k_j \le m$, we have

$$\pi_{a_j^{\ell}} \le \frac{1}{V - LU_L + e^{-1}LM}.$$

Similarly, for $v \notin \mathcal{U}$, $\pi_v = Z^{-1} \le (V - LU_L + e^{-1}LM)^{-1}$. Using the assumption on M finishes the proof. □

Lemma C.2. *Under the same conditions as Theorem 1, assuming $\alpha_2 \le (\lambda_1 + m\lambda_2)^{-1}$, when using the weights,*

$$\boldsymbol{W}_{KQ}^{\textit{subj}} = \beta\Big(\sum_j \boldsymbol{h}(s_j) + \alpha_1 \boldsymbol{h}(p_{-1})\Big)\boldsymbol{h}([\textit{sep}])^\top, \qquad \boldsymbol{W}_{OV}^{\textit{subj}} = -\sum_j \Big(\sum_{j'\neq j,\ell} \boldsymbol{h}(a_{j'}^{\ell})\Big)\boldsymbol{h}(s_j)^\top,$$

$$\boldsymbol{W}_{KQ}^{\textit{rel}} = \beta\Big(\sum_{\ell=1}^{L} \boldsymbol{h}(r_\ell) + \boldsymbol{p}\Big)\boldsymbol{h}([\textit{sep}])^\top,$$

$$\begin{aligned} \boldsymbol{W}_{OV}^{\textit{rel}} = &\sum_{\ell}\Big(\sum_j \boldsymbol{h}(u_j^{\ell})\Big)\boldsymbol{h}(r_\ell)^\top + \alpha_2\Big((m-1)\sum_{\ell=1}^{L}\Big(\sum_{a=1}^{U_L}\boldsymbol{h}(u_a^{\ell})\Big)\Big(\sum_{a=1}^{U_L}\boldsymbol{h}(u_a^{\ell})\Big)^\top \\ &+ \frac{U_L^2}{M_{\textit{ft}}}\sum_{j=1}^{M_{\textit{ft}}}\Big(\sum_{\ell=1}^{L}\boldsymbol{h}(a_j^{\ell})\Big)\boldsymbol{h}(s_j)^\top\Big), \end{aligned}$$

the prediction probabilities π_v are bounded as:

$$\pi_v \le \frac{e}{V + LU_L(e^{-1} - 1)} \le \frac{e}{M}.$$

Proof. Let $m := N_{\text{ft}} + 1$ denote the number of subject positions in the ICL sequence, and let k_j be the multiplicity of subject j among these positions (so $\sum_j k_j = m$), and define $\mathcal{S}' := \{j : k_j > 0\}$.

When $\beta \to \infty$, the relation head attends uniformly over all T positions, while the subject head attends to the query subject $\boldsymbol{h}(s_{j_{N_{\text{ft}}+1}})$. For any attribute token $v = a_j^\ell$, define

$$c_j(\boldsymbol{X}) := |\{\, i \in [N_{\text{ft}} + 1] : j_i = j \,\}|,$$

i.e., the number of times subject j appears among the subjects in the given sequence $\boldsymbol{X}$. Then

$$\boldsymbol{h}(a_j^\ell)^\top f(\boldsymbol{X}) = \alpha_2 \left((m-1)\frac{N_{\text{ft}}}{\mathrm{T}} \mathbb{1}\left[\ell = \ell_\star\right] + \frac{U_L^2}{M_{\text{ft}}\mathrm{T}} c_j(\boldsymbol{X}) \right) - \mathbb{1}\left[j \neq j_q\right].$$

Let $Z := \sum_{u \in \mathcal{V}} \exp(\boldsymbol{h}(u)^\top f(\boldsymbol{X}))$ and let

$$\pi_v = \frac{\exp(\boldsymbol{h}(v)^\top f(\boldsymbol{X}))}{Z}.$$

Let $\lambda_1 := (m-1)N_{\text{ft}}/\mathrm{T}$, $\lambda_2 := U_L^2/(M_{\text{ft}}\mathrm{T})$. Then, we have

$$\pi_v \leq \frac{\exp(\alpha_2(\lambda_1 + m\lambda_2))}{U_L(\exp(\alpha_2(\lambda_1+\lambda_2)-1) + (L-1)\exp(\alpha_2\lambda_2 - 1)) + V - LU_L} \leq \frac{e}{V + LU_L(e^{-1}-1)},$$

since $\alpha_2 \leq (\lambda_1 + m\lambda_2)^{-1}$.

□

C.3 Experimental Validation

C.3.1 Relative Positional Encoding

In this section we present some additional experimental results for experimental validation in Section 4.

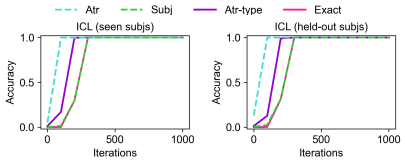

Figure 18: Finetuning the single-layer attention-only model with relative positional encoding pretrained on PT sequences using ICL sequences with a subset of subjects enables generalization on ICL sequences with held-out query subjects.

Validation of ICL Mechanism. We finetune the model on ICL sequences (Eq. (3)) with the last-token prediction objective, using 50% of the total subjects (Fig. 18 shows that the model generalizes on held-out subjects). Visualizing the attention scores on an ICL sequence in Fig. 19 reveals that the model repurposes its heads for the new task, consistent with Prop. 2. We find that the head 2 (subject head) attends to the most recent/query subject token, while head 0 (relation head) attends to the attribute tokens in the context. Further, Fig. 20 confirms that the outputs of these heads closely match the theoretical outputs specified by our construction for ICL sequences: head 0 (subject head) outputs negative sum of all attributes not associated with the subject $s_{j_{N_{\text{ft}}+1}}$ (left subplot), while head 1 (relation head) outputs the sum of all attributes of type $\ell_\star$ (right).

In Figs. 21 and 22, we visualize a sliced version of attention weights $\boldsymbol{W}_{OV}^{\text{subj}}$ (left) and $\boldsymbol{W}_{OV}^{\text{rel}}$ (right) after pretraining and finetuning, respectively. Specifically, rowwise or columnwise, the first 8 entries correspond to subjects, the next 8 to attribute type $\ell = 1$ tokens, the next 8 to $\ell = 2$, the next 5 to grammar and separator tokens, and the final 8 to the relation tokens. From the two figures, we observe that the subject heads encode information about subject-attribute associations by suppressing attributes not associated with the subject, and don't change much after finetuning. On the other hand, the relation head initially encodes information about relation type-attribute associations, but after finetuning, it changes significantly, encoding attribute-attribute associations for the same attribute type. This is consistent with our constructions in Section 4.

In Fig. 23, we observe that for larger number of PT subjects, the performance is overall lower. To examine why this is the case, focusing on the setting with 32 FT subjects with larger number of unique attributes, in Fig. 24, we show i) after pretraining, the cosine similarities of the output of each head with the subject and relation head outputs predicted by the construction in Proposition 1, and ii)

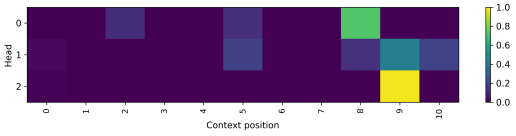

Figure 19: Attention scores for each head across an ICL sequence at the end of finetuning. Head 2 attends to the most recent subject, while head 0 attends to the attribute tokens in the sequence, as predicted by Proposition 2.

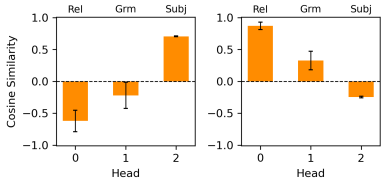

Figure 20: Cosine similarity between the actual outputs of each head and the outputs $f_{\text{subj}}(\boldsymbol{X})$ (left) and $f_{\text{rel}}(\boldsymbol{X})$ (right) from our construction in Proposition 2. Head 2 output closely matches $f_{\text{subj}}(\boldsymbol{X})$, while head 0 matches $f_{\text{rel}}(\boldsymbol{X})$.

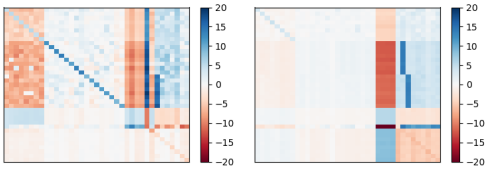

Figure 21: Visualization of the weight matrices $\boldsymbol{W}_{OV}^{\text{subj}}$ (left) and $\boldsymbol{W}_{OV}^{\text{rel}}$ (right) after pretraining the single-layer attention-only model on PT sequences.

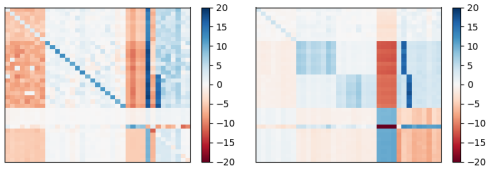

Figure 22: Visualization of the weight matrices $\boldsymbol{W}_{OV}^{\text{subj}}$ (left) and $\boldsymbol{W}_{OV}^{\text{rel}}$ (right) after finetuning the single-layer attention-only model on ICL sequences.

after FT, the performance on ICL sequences with held-out query subjects. We observe the following. First, for 256 PT subjects, the cosine similarities for subject and relation head are high, and after FT, the subject-match accuracy is high, while attribute-type match accuracy is lower because all attributes were not seen during FT. On the other hand, for 2048 PT subjects, the cosine similarities for subject and relation head are low, and after FT, the subject-match accuracy is low, as well as the attribute-type match accuracy is low because all attributes were not seen during FT.

Interestingly, this suggests that representations learned during pretraining can affect downstream generalization performance after finetuning, even though in both cases, the model achieves perfect accuracy for factual recall as well as on ICL sequences with seen subjects. We believe that a deeper investigation of this phenomenon including how to improve the representations learned after pretraining is an interesting direction for future work.

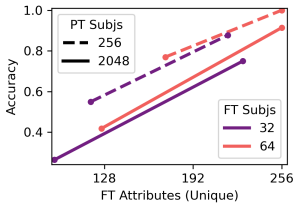

Figure 23: Effect of number of unique attributes seen during finetuning (with ICL sequences) on the accuracy on ICL sequences with held-out query subjects, across different number of PT and FT subjects. Performance improves as the number of unique attributes seen while finetuning increases.

C.3.2 Absolute Positional Encoding

In this section, we present additional experimental evidence to corroborate our theoretical constructions, using a 1-layer 3-head attention-only model with absolute positional encodings (see App. B for details).

Validation of Pretraining Mechanism. We first train the model using PT sequences (Eq. (2)) with the next-token prediction objective. In Fig. 25, we visualize the attention scores for each head across a PT sequence, and find that the heads specialize into distinct roles consistent with Proposition 1. We find that head 0 attention score concentrates on the first subject token, and it outputs $\boldsymbol{h}(s_{\bar{j}})$, matching the role of $g_{\text{subj}}(\boldsymbol{X})$ in our construction, while head 1 attends to the most recent relation token, *i.e.*, it outputs $\boldsymbol{h}(r_{\ell_{N_{\text{tr}}}})$, consistent with $g_{\text{rel}}(\boldsymbol{X})$. Further, in Fig. 26, we compute the cosine similarity between the head outputs and theoretical outputs of the subject and relation heads specified by our construction. Specifically, we report the cosine similarity with $f_{\text{subj}}(\boldsymbol{X})$, *i.e.*, negative sum of all attributes not associated with the subject $s_{\bar{j}}$ (left subplot) and $f_{\text{rel}}(\boldsymbol{X})$, *i.e.*, sum of all attributes of type $\ell_{N_{\text{tr}}}$ (right), averaged across several sequences. We find that head 0 output closely matches $f_{\text{subj}}(\boldsymbol{X})$, while head

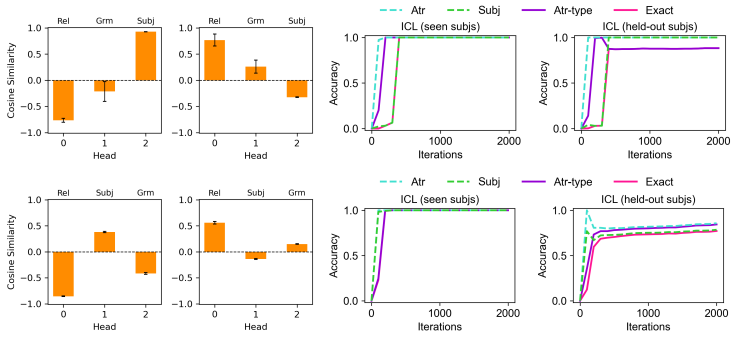

Figure 24: Using 256 (top row) or 2048 (bottom row) PT subjects, (left) after pretraining, the cosine similarities of the output of each head with the subject and relation head outputs predicted by the construction in Proposition 1, and (right) after FT with 32 subjects, the performance on ICL sequences with held-out query subjects. In the first case, the cosine similarities for subject and relation head are high, and after FT, the subject-match accuracy is high, while attribute-type match accuracy is lower because all attributes were not seen during FT. In the second case, the cosine similarities for subject and relation head are low, and consequently, after FT, the subject-match accuracy is low, as well as the attribute-type match accuracy is low because all attributes were not seen during FT.

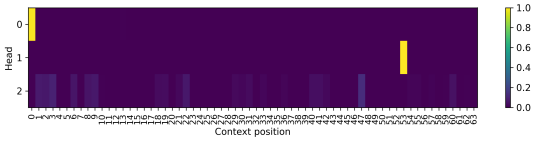

Figure 25: Attention scores for each head across a PT sequence at the end of pretraining. Head 0 attends to the first subject, while head 1 attends to the most recent relation token, as predicted by Proposition 1.

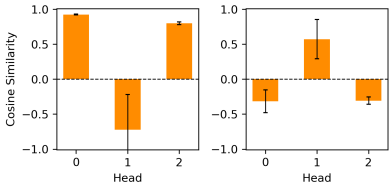

Figure 26: Cosine similarity between the actual outputs of each head and the outputs $f_{\text{subj}}(\boldsymbol{X})$ (left) and $f_{\text{rel}}(\boldsymbol{X})$ (right) from our construction in Proposition 1. Head 0 output closely matches $f_{\text{subj}}(\boldsymbol{X})$, while head 1 matches $f_{\text{rel}}(\boldsymbol{X})$.

1 matches $f_{\text{rel}}(\boldsymbol{X})$. We designate these heads as subject and relation heads, respectively. Together, these results present experimental validation of our construction for PT sequences.

Validation of ICL Mechanism. We finetune the model on ICL sequences (Eq. (3)) with the last-token prediction objective, using 50% of the total subjects (Fig. 27 shows that the model generalizes on held-out subjects). Visualizing the attention scores on an ICL sequence in Fig. 28 reveals that the model repurposes its heads for the new task, consistent with Prop. 2. We find that the head 0 (subject head) attends to the most recent/query subject token, while head 1 (relation head) attends to the attribute tokens in the context, *i.e.*, it outputs a combination of attributes of type $\ell_\star$. Further, Fig. 29 confirms that the outputs of these heads closely match the theoretical outputs specified by our construction for ICL sequences: head 0 (subject head) outputs negative sum of all attributes not associated with the subject $s_{j_{N_{\text{ft}}+1}}$ (left), while head 1 (relation head) outputs the sum of all attributes of type $\ell_\star$ (right).

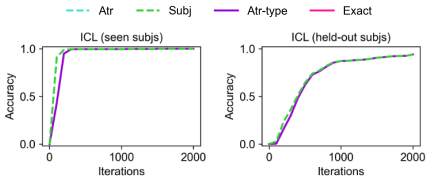

Figure 27: Finetuning the single-layer attention-only model with absolute positional encoding pretrained on PT sequences using ICL sequences with a subset of subjects enables generalization on ICL sequences with held-out query subjects.

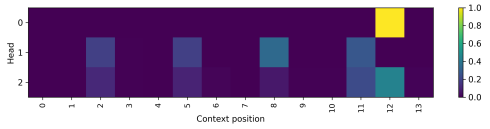

Figure 28: Attention scores for each head across an ICL sequence at the end of finetuning. Head 0 attends to the most recent subject, while head 1 attends to the attribute tokens in the sequence, as predicted by Proposition 2.

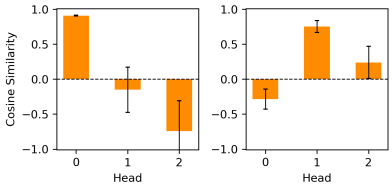

Figure 29: Cosine similarity between the actual outputs of each head and the outputs $f_{\text{subj}}(\boldsymbol{X})$ (left) and $f_{\text{rel}}(\boldsymbol{X})$ (right) from our construction in Proposition 2. Head 0 output closely matches $f_{\text{subj}}(\boldsymbol{X})$, while head 1 matches $f_{\text{rel}}(\boldsymbol{X})$.

D LIMITATIONS

For our experiments, we use small-scale transformer models and a synthetic data setting to enable controlled analysis. This poses the question of whether our findings generalize to LLMs used in practice. While our setup abstracts away the richness of natural language, this abstraction provides precise control over the pretraining distribution, which is standard in other works on understanding ICL, where small-scale tasks (linear regression (Garg et al., 2022; Raventos et al., 2023; von Oswald et al., 2022), Markov chains (Edelman et al., 2024a; Park et al., 2025)) are used to probe task diversity, training dynamics, etc. In the same vein, our work aims to understand when and how contextual recall emerges. As LLMs are pretrained on internet scale data, it makes it challenging to identify the role of different types of data in helping the model develop specific abilities. Our synthetic setup enables controlled analysis of how different types of data enable emergence of the contextual recall ability. Notably, despite its minimalism, the framework reproduces key phenomena observed at larger scales, including stage-wise learning (Zucchet et al., 2025) and the formation of task vectors (Hendel et al., 2023; Todd et al., 2024). We see our work as a first step toward mechanistically understanding contextual recall as a well-defined and practically relevant form of ICL, and leave broader generalization as future work.

For our theoretical analysis, we make certain assumptions which enable tractability but also may not be realistic. Specifically, we consider a simpler setting where the attribute type information is encoded in a dedicated relation token, and use fixed orthogonal token embeddings and positional encodings. Our analysis of the finetuning dynamics assumes some of the parameters to be fixed, and uses one or two steps of gradient descent for other parameters. While these assumptions enable tractable analysis, extending the theoretical results to more general settings is also an important direction for future work.