Memorization in Attention-only Transformers

Léo Dana[1], Muni Sreenivas Pydi[1], and Yann Chevaleyre[1]

[1]MILES, Universitée Paris-Dauphine.

Abstract

Recent research has explored the memorization capacity of multi-head attention, but these findings are constrained by unrealistic limitations on the context size. We present a novel proof for language-based Transformers that extends the current hypothesis to any context size. Our approach improves upon the state-of-the-art by achieving more effective exact memorization with an attention layer, while also introducing the concept of approximate memorization of distributions. Through experimental validation, we demonstrate that our proposed bounds more accurately reflect the true memorization capacity of language models, and provide a precise comparison with prior work.

1 Introduction

Modern large language models, especially Transformers, showcase great memorization capacity [7, 13]. Among recent works, researchers have shown that facts are memorized in the MLPs of a Transformer, and have even identified which MLPs [10, 11, 12]. However, they were not able to understand *how* these MLPs store information. Both exact and approximate theoretical memorization in an MLP are well documented in the literature: a ReLU MLP can memorize exactly as many real-valued label as it has neurons n [3], and can memorize n discrete labels with only $\tilde{O}(\sqrt{n})$ neurons [14].

Contrary to the MLPs, the memorization power of multi-head attention layers has not been empirically studied. The main role of the attention layer is not viewed as remembering information but rather as moving between residual streams the information retrieved by the MLPs [12, 16, 15]. For the theoretical aspect of the memorization in attention layers, there exists results on the expressivity of the attention patterns [1], the memorization capacity of attention layers [9], and the memorization capacity of Transformers [8]. We will discuss related works in depth in section 6.

In this article, we are interested in moving the state-of-the-art in terms of memorization capacity for the attention layer. We will thus consider the memorization capacity of an Attention-only Transformer (AoT). We need to specify what *memorization capacity* means. We will distinguish two types of memorization tasks, namely the association task and the distribution task.
The **association** task, already studied in [2, 4, 8, 9], consists of predicting a token given a sequence of tokens as input. We only require the AoT to predict the right next-token at the last position. This memorization is exact and hence, we want to know the maximal set of sequence-token associations (t_{in}, t_{out}) that can be exactly memorized by an AoT.
The **distribution** task consists of predicting the correct distribution, measured using the KL-divergence, for an input sequence of tokens. We use the KL-divergence since it is the default loss function used to train most Transformers. To our knowledge, we are the first to introduce and study this task. emorizing distribution happens in natural language modelisation: take the sentence "Arnold Schwarzenegger was a", it can be completed with "actor", "writer" or "culturist". Thus, language models need to memorize not one but several correct next-tokens, each with a possibly different probability depending on the importance of the answer.

Our contributions are:

- We improve the state-of-the-art on the association task by proving that a one layer AoT with H heads each of dimension d_h, and an embedding dimension d can memorize $Hd_h + d$ associations. In the context of language model, this improves on the previous result on the attention layer expressivity by [9] which requires a limited context-windows and has memorization capacity of $H(d_h-1)+1$. We compare our result with other constructions using deep Transformers [8] as well as MLPs memorization [6, 3].

- We introduce the distribution task for Transformers as a way to quantify memorization when there is not a unique correct next-token. We provide upper and lower bounds for that distribution task on the error made by the best AoT. The divergence of that AoT will approximate that of a sequence encoder, which is a mapping from token sequences to logits, and has a rank constraint.
- Finally, we prove upper bound for the divergence of sequence encoders when the target distribution is almost a look-up table.

Proofs for the statements can be found in appendix with experimental details. The code-base is available at this repository.

2 Formalism

We study an Attention-only Transformer (AoT) that has only one layer of multi-head attention (MHA) mechanism denoted by $\mathcal{A}$. Let $[N]=\{1,...,N\}$ be the token dictionary. Each token is embedded in dimension d by the embedding $e:[N]\to\mathbb{R}^d$, and a positional embedding is added based on its position s. Thus, we denote a sequence of S tokens by $t_{1:S}$. We also denote t_{S+1} for the output token. The MHA contains H heads, each of inner dimension d_h, with $d_h\leq d$[1], meaning that $W_Q^h, W_K^h, W_V^h \in \mathbb{R}^{d_h,d}$ and $W_O^h\in\mathbb{R}^{d,d_h}$. Following the intuition from [5], we choose to separate output matrices in each head, as well as combine $W_{QK}^h=(W_Q^h)^TW_K^h$. Each attention head can be written as follows.

$$\begin{aligned}\mathcal{A}^h(t_{1:S}) &= W_O^hW_V^hA^h(t_{1:S}) = W_O^h\sum_{s=1}^{S}a^h(t_{1:S})_s(e(t_s)+pos_s)\\ a^h(t_{1:S}) &= \text{Softmax}((e(t_S)+pos_S)^TW_{QK}^h(e(t_s)+pos_s), s=1:S)\end{aligned}\tag{1}$$

We concatenate the output matrices into $W_O\in\mathbb{R}^{d,Hd_h}$ and the attention before output $A(t_{1:S})\in\mathbb{R}^{Hd}$. We also construct the matrice $W_V\in\mathbb{R}^{Hd_h,Hd}$ as block diagonal with block W_V^h, which has full rank when each W_V^h has full rank. Then, the output of the attention layer is added to the residual stream and goes through an unembedding matrix $W_U\in\mathbb{R}^{N,d}$ to obtain logits for the next token. Since we are only interested in the next-token prediction at the last position, we denote the AoT's computation by

$$\mathcal{T}(t_{1:S}) = W_U\left(e(t_S)+pos_S+W_OW_VA(t_{1:S})\right)\tag{2}$$

For both the association and distribution tasks, we define the conditional distribution $\pi_{t_{1:S}}$ and a prior distribution π over token sequences $t_{1:S}$. The task of the Transformer is to minimize the KL-divergence with the conditional distribution for each input sequence, averaged over the prior distribution.

$$d_{KL}(\pi,\mathcal{T}) := \mathbb{E}_{t_{1:S}\sim\pi}[d_{KL}(\pi_{t_{1:S}}||\text{Softmax}\circ\mathcal{T}(t_{1:S}))]$$

The association case is a restriction of the distribution case to conditional distibutions with 0 entropy, which is equivalent to having one next-token of probability 1. We denote this setting as assumption 1 below, and we will use it when refering to the association task.

Assumption 1. *For all sequence token $t_{1:S}$, there exists t_{S+1} such that $\pi(t_{S+1}|t_{1:S})=1$. This is equivalent to π having a conditional distribution with 0 entropy.*

In the association case, we say that the Transformer memorizes an example $(t_{1:S},t_{S+1})$ if $\mathcal{T}(t_{1:S})_{t_{S+1}}$ is the maximum logit, and we let T_0 the number of sequence-token association to memorize, which is at most N^S. In the distribution case, the Transformer memorizes example $t_{1:S}$ if $\mathcal{T}(t_{1:S})=\log(\pi_{t_{1:S}})$. We introduce another assumption that arises in the distribution case.

Assumption 2. *For all $t_{1:S},t_{S+1}$, $\pi(t_{S+1}|t_{1:S})\neq 0$. This is equivalent to π having a conditional distribution with full support..*

[1] Since matrices W_O, W_V and W_Q, W_V are multiplied together, taking $d_h\leq d$ will decrease their rank to d_h, but taking $d_h>d$ will simply have them full rank, which is suboptimal compared to taking $d=d_h$.

3 Memorization limit of Transformer

Looking at equation (2), we see that the AoT can be written as $\mathcal{T}(t_{1:S}) = W_U E(t_{1:S})$ where $E(t_{1:S}) = e(t_S) + pos_S + W_O W_V A(t_{1:S})$ is a sequence embedding (rather than a token embedding). Our AoT belongs to the set of **sequence encoders** defined below.

Definition 1. *Let $\mathcal{L}(N,S,d) = \left\{f_{W,E} | W \in \mathbb{R}^{d,N}, E : [N]^S \to \mathbb{R}^d\right\}$ with $f_{W,E}(t_{1:S}) = WE(t_{1:S})$ the set of maps that embed token sequence and unembed them into logits. We call them sequence encoders and let*

$$d_{KL}\left(\pi, \mathcal{L}(N,S,d)\right) := \inf_{f \in \mathcal{L}(N,S,d)} d_{KL}(\pi, f).$$

In full generality, Transformers with any number of MLPs or attention layers are sequence encoders. Indeed, one can think about every computation happening before the unembedding as a sequence embedding parametrized by few parameters (in comparison to unconstrained sequence embdedding). Thus, as stated in Proposition 1 below, Transformers can memorize distribution at most as well as the best sequence encoders.

Proposition 1. *Let $\mathcal{T}$ be any Transformer with embedding dimension d, dictionary size N and context window S, and π be any distribution, we have*

$$d_{KL}(\pi, \mathcal{T}) \geq d_{KL}\left(\pi, \mathcal{L}(N,S,d)\right). \tag{3}$$

In particular, as stated in Proposition 2, when $d < N - 1$, the infimum can be non-zero. This creates a rank d bottleneck for the memorization of distributions by the Transformer. This means that in general one cannot have a Transformer remember more than d distributions exactly. As we will see in Theorem 1, approximate memorization is more suitable to the distribution task, as measured by the distance to this lower-bound.

Proposition 2. *For any distribution π, if $d \geq N - 1$, then $d_{KL}\left(\pi, \mathcal{L}(N,S,d)\right) = 0$. Conversely, if $d < N - 1$, there exists a distribution π such that $d_{KL}\left(\pi, \mathcal{L}(N,S,d)\right) > 0$.*

For the association task, the bottleneck doesn't exist in any embedding dimension d. The appropriate way to evaluate memorization in the association task is using exact memorization.

Proposition 3. *Under Assumption 1, for $d \geq 2$, $d_{KL}\left(\pi, \mathcal{L}(N,S,d)\right) = 0$.*

While sequence encoders don't have limit to their associative memorization capacity, under-parametrized AoT do. We give an upper bound on the memorization capacity of a one-layer AoT in Corollary 2, and we compare it to experimental scaling laws in section 4.2.

4 Memorization capacity of AoT

In this section, we present our main results that respectively give upper bounds on memorization in the distribution and association settings. We will start with the result on remembering distributions, making the association task a corollary.

Define T_ε as the smallest number of token sequences whose cumulative probability is greater than $1-\varepsilon$. We have $T_\varepsilon \leq \left\lceil (1-\varepsilon)N^S \right\rceil$, the upper bound being attained when the probability distribution over token sequences is uniform. This notation is consistent with T_0 defined earlier, the number of non-zero probability sentences. The theorem below states that we can construct a Transformer which approximates the lower bound set in Proposition 1 arbitrarily.

Theorem 1. *Let $\varepsilon \geq 0$ and $\gamma > 0$. Under Assumption 2 there exists $f_{W,E}$ and an AoT $\mathcal{T}$ with embedding dimension d, head dimension d_h, and H attention heads, satisfying $d_h H + d \geq T_\epsilon$, such that*

$$d_{KL}(\pi, \mathcal{T}) \leq d_{KL}(\pi, \mathcal{L}(N,S,d)) + C\varepsilon ||WE||_2 + \gamma \tag{4}$$

$\mathcal{T}$ has $d(S + 2N + 4d_h H)$ parameters, and C depends only on A, the attention before output.

Remarks 1.

- In the parameter count, dS and dN are necessary for the word embedding, positional embedding and unembedding. The $dd_h H \simeq dT_\varepsilon - d^2$ scaling comes from the attention heads and is the comparison point with previous work as we will detail in section 6.

- In practice, sequence-token pairs don't have the same probability. Thus, our AoT remembers all most likely sequences up to ε, which explains that we are close to the lower bound for small epsilon. The term $||WE||_2$ corresponds to the worst possible prediction, and is standard in the literature on MLP expressivity[2]. The term in γ is simply a term that can be taken infinitely close to 0, and accounts for approximating the skip connection with a special attention head.

- Our bound doesn't leverage the particularity of the attention architecture, only the high rank of the attention mechanism. It is still an open challenge to provide better bounds using the attention properties, as for ReLU MLPs, especially, to upper bound C.

- One can take $\varepsilon = 0$ to achieve the lower bound set in Proposition 1 with $Hd_h + d = T_0$ attention heads, and uses $d(S + 2N + 4(T_0 - d))$ parameters. In that case, one could say that the Transformer remembers exactly the distributions, even if the divergence is not 0, since it is the smallest loss attainable by the Transformer architecture.

We can also use our Theorem 1 under Assumption 1: we use it on the probability $\pi_\delta = \frac{\pi+\delta}{1+N\delta}$ with $\delta > 0$ small enough that satisfy Assumption 2, such that the Transformer has perfect accuracy for π, with $\gamma > 0$ small. In this association task, the AoT thus remembers T_0 associations.

Corollary 1. *Under Assumption 1, there exist an AoT $\mathcal{T}$ with embedding dimension 2, H attention heads and d_h inner head dimension, which can memorize at least $T_0 = Hd_h + 2$ associations, using $2(S + 2N + 4(T_0 - 2))$ parameters.*

4.1 Sketch of the Proof

We present below the proof of our main result. We prove first the case $\varepsilon = 0$ as it is useful for the $\varepsilon > 0$ case. We work under Assumption 2 in both cases in order to use Lemma 1 below.

Lemma 1. *Under Assumption 2, there exists $f \in \mathcal{L}(N, S, d)$ such that $d_{KL}(\pi, f) = d_{KL}(\pi, \mathcal{L}(N, S, d))$*

1. Case $\varepsilon = 0$: we have our AoT as in equation (2), and thanks to Lemma 1 we take $f_{W,E}$ an optimal sequence encoder that we approximate. By choosing $W_U = W$, we want to solve the system $E(t_{1:S}) = e(t_S) + pos_S + W_O W_V A(t_{1:S})$ for all token sequence $t_{1:S}$.

We start by showing that the skip connection $e(t_S) + pos_S$ can almost be written as an attention head, by changing a change of basis $e(t_S) + pos_S = W_O^0 W_V^0 (e'(t_S) + pos'_S)$ and using the attention pattern $W_{QK}^0 = \lambda I_d$. When λ is large enough, the attention head with matrices W_{QK}^0, W_V^0, W_O^0 becomes arbitrarily close to $e(t_S) + pos_S$. Thus, we can augment A', the attention before output, by adding head 0, which has an inner dimension of d. Since the result is only valid in the limit of $\lambda \to +\infty$, this explains the constant $\gamma > 0$ in the theorem.

Now, $W_V A' \in \mathbb{R}^{Hd_h+d, T_0}$, and since $T_0 \leq Hd_h + d$, the linear system $E(t_{1:S}) = W_O W_V A'(t_{1:S})$ is solvable, where the variable is W_O, if the family $\{W_V A'(t_{1:S})\}_t$ has rank T_0, which is equivalent to having $\{A'(t_{1:S})\}_t$ with rank T_0 since W_V reduces the rank. We are left with proving Lemma 2.

Lemma 2. *Let A the attention before output of a model with H heads. For T_0 token sequences, there exists matrices W_{QK}^i and embeddings e and pos such that the family $\{A(t_{1:S})\}_t$ has rank greater than* $\min(T_0, Hd)$.

First, one can see that it is sufficient to prove the Lemma when $T_0 \geq Hd$, since otherwise one can add dummy examples to have $T_0 = Hd$. We compute the rank in terms of the rows, head by head, meaning that Lemma 2 reformulates as proving that the function $W_{QK}^h \to A^h$ has its image not contained in any hyperplane of $\mathbb{R}^{d,T_0}$.

This is now a technical result: recall that an attention head output is a weighted average of the tokens by the attention pattern. Now, suppose that this function is orthogonal to some vector $v \in \mathbb{R}^{d,T_0}$ for all matrices W. Using Taylor expansions, we transform the softmax from the attention pattern into infinite sums of exponential. The equation is now stating that exponential functions sum to 0 for all W. Thus, by recognizing that exponential functions form a free family, we identify the coefficients for each different exponent. Using handcrafted exponent, we show that the coefficients before the exponents form a system, the only solution of which is $v = 0$.

Remark: we see that in the statement of Lemma 2, one doesn't need to have ...

[2] See Theorem 2 in [3].

2. Case $\varepsilon > 0$: here, we reuse the previous idea on the most likely sequences that cumulate $1-\varepsilon$ of the probability. So we solve the same linear system as before for the most likely sequences, and we control the error for the low probability sequences. If we note S_2 the set of the low probability sequences, we can bound the difference in KL-divergence by the L_2 norm of the logits

$$|d_{KL}(\pi, f_{W,E}) - d_{KL}(\pi, \mathcal{T}^*)| \leq \mathbb{E}_{t_{1:S} \in S_2}\left[|WE(t_{1:S}) - \mathcal{T}(t_{1:S})|\right] \leq \varepsilon ||WE - \mathcal{T}||_2$$

With the same choice as in the first case, we can bound the L_2 norm of the logits difference by the term $C||WE||_2$ which represent the error of the best predictor, and the constant in the theorem is $C = \sqrt{1 + ||(P_1VP_1^T)^{-1}||_2^2}$ where P_1 is the projection that keeps the most likely sentences, and $W_VA' = UI_\Sigma P_1V$ the singular value decomposition of W_VA'[3]. The constant C is the term that depends on the expressivity of the attention mechanism itself and is thus hard to control with only Lemma 2.

natural question is whether one can drop Assumption 2 ? To do so, one can take a sequence encoder f_δ which is approximating the minimum divergence at δ, and whose limit distribution is π_0, and try to approximate f_δ using the AoT. Thus, one has to control the growth of $||f_\delta||_2$. But the assumption on f_δ is not enough to control its growth : it just ensures that the convergence is fast enough, but we need an upper bound on the speed of convergence ! Thus, we need $c > 0$ and $\alpha \geq 1$ such that for any token sequences t satisfying $\pi_0(t_{S+1}|t_{1:S}) > 0$, we have the property

$$c\delta^\alpha \leq \left|\log\left(\frac{\pi_0(t_{S+1}|t_{1:S})}{\text{softmax}(f_\delta(t_{1:S}))_{t_{S+1}}}\right)\right| \leq \delta \tag{5}$$

The lower bound will be used to prove that the norm of f_δ can be controled by $O\left(\log\left(\frac{1}{\delta}\right)\right)$, while the upper bound gives the convergence of f_δ to the desired distribution. We develop this idea in Appendix A.3 to prove an weaker alternative to Theorem 1 which holds for any π.

4.2 Memorization scaling laws

In Corollary 1, we obtained a lower bound on the number of associations a one layer AoT can remember. We now want to know what the empirical scaling of the memorization power is. To this end, we will analyze scaling laws of the accuracy of trained AoT for different values of H, d, and d_h.

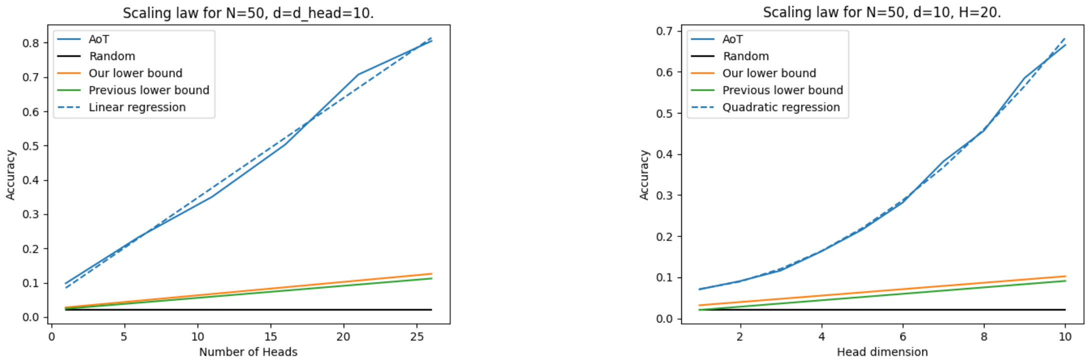
In our experiments, we train an AoT on bigrams, meaning that $S = 2$, and with a dictionary size of $N = 50$. Since we want to measure associative memory, we take π satisfying Assumption 1. Moreover, we take the prior distribution of π uniform over all pairs of tokens. This way, the accuracy will be a good measure of the associative memory. But accuracy measures the performance of the AoT in average, meaning that by pure chance, using 0 head, the AoT will have $\frac{1}{N}$ accuracy, and our Corollary 1 is a statement in worst-case. Thus, to compare the scaling laws in accuracy with our results, we first need to state Corollary 2.

Corollary 2. *Under Assumption 1, there exist an AoT* $\mathcal{T}$ *such that* $\mathbb{E}_{t_{1:S}}[Acc(\mathcal{T}(t_{1:S}))] \geq \frac{1}{N} + \left(1 - \frac{1}{N}\right)\frac{Hd_h + d}{T_0}$.

In general, the transformation $\phi : X \in [0, T_0] \to \frac{1}{N} + (1 - \frac{1}{N})\frac{X}{T_0} \in [0, 1]$ gives the correspondance between associative memory and accuracy. We will use Corollary 2 as a lower bound for our experiment, which we call *Our lower bound.* The other lower bound called *Previous lower bound* is the one obtained in [9], which we improve on, and discussed in section 6. For more details on the experiments, please read appendix B.

As said before, we want to understand the empirical scaling laws of the associative memory. In particular, we are interested in the following questions: is the scaling linear in H and d_h ? what is the effect of d on the scaling laws ? To answer the first question, we trained AoT with $d = 10$ fixed, we vary H and d_h, and we measure the accuracy after training. Then, we plot the scaling laws when H grows with a constant d_h. Both lower bound for the memorization power of the AoT are displayed, along with the baseline accuracy of a random AoT.

[3] Here, the matrices V and W_V denote different object and are not subscript from one another. We choose to keep the usual notations from the singular value decomposition as well as the attention mechanism.



(a) Accuracy scaling law as H grows. The head dimension is fixed to 10. The scaling is linear.

(b) Accuracy scaling law as d_h grows. The model has 20 heads. The scaling is quadratic.

Figure 1: Scaling laws on H and d_h. The embedding dimension is 10 and the dictionnary size is 50. The blue dotted lines are the linear or quadratic least square approximation of the empirical accuracy.

As observed on figure 1.a, the scaling in H is linear for d_h fixed, and on 1.b the scaling in d_h is quadratic when H is fixed. This means that the memorization power empirically scales as $C(d,N)Hd_h^2 + C'(d,N)$. Although the scale is linear in H as in Corollary 1, the constant is not the same as the growth is actually much faster. The comparison cannot be thight since Corollary 1 gives the same scaling in H for all $d \geq 2$[4]. In Appendix B, we test the scaling of H with $d = 2$. Figure 4 shows that even in this case, the empirical scaling law is around twice as fast.
The constant term $C'(d,N)$ is proven to be linear in d in [4], meaning that a matrix multiplication of rank d can store at most d associations. Interestingly, this experiment shows that taking $d = d_h$ is optimal in term of memorization per parameter, since the parameter scaling is linear in d_h yet quadratic (for $d_h \leq d$) in memorization power. This quadratic scaling seems to be linked to the increase in expressivity of the attention pattern rather than the increase in the rank. Indeed, the rank increase when the number of heads increases only leads to a linear increase in the memorization capacity.

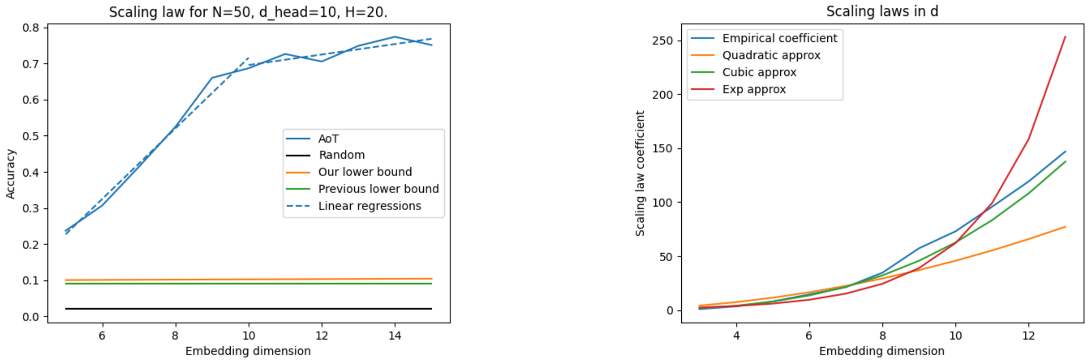
Next, we try to understand the scaling of $C(d,N)$ in the variable d. First, we fix $H = 20$ and $d_h = 10$ to plot the scaling law. As shown on figure 2.a, the scaling law is composed of two parts: $d \leq d_h$ and $d \geq d_h$. When $d \geq d_h$, the scaling is expected to be linear since only the term $C'(N)d$ should grow, because the increase in embedding space is not beneficial to the heads since they have smaller inner dimension. The fact that we observe a greater linear scaling for $d \leq d_h$ also suggest that $C(d,N)$ is linear in d for $d \leq d_h$ only.
Second, we compute the scaling of $C(d,N)$ using $d = d_h$. This way, we can compute $C(d,N)d_h$ by measuring the growth of the accuracy with H for different values of d. Figure 2.b shows the growth of this scaling coefficient and compares it to different models. Each model is computed to be the least-square approximation. You can find more details in Appendix B. We observe that the coefficient has a cubic growth when $d = d_h$, leading to the scaling $C(N)d^3H$. These two results coïncide when $C(d,N) = C(N)d$ for $d \geq d_h$.

4.2.1 MLP memorization

The motivation for this paper was to understand the memorization power of the attention mechanism, especially in term of associative memory. We now know that the attention mechanism can memorize associations, but it is still unkown how memorizing with attention heads compares to memorizing with an MLP. To understand whether attention heads or MLPs will be memorizing in LLMs, we want to compare the accuracy scaling laws of an MLP-based Transformer with that of an AoT with equal number of parameters. To that end, the AoT will be taken as before with $d = d_h$, and the MLP-based Transformers will be one layer of attention with only one head of dimension $d_h = d$ as well, followed by an MLP layer with width w.

$$\mathcal{T}(t_{1:S}) = W_U(W_2\sigma W_1 + I_d)(W_O W_V A(t_{1:S}) + e(t_S) + pos_S)$$

[4] This is all the more true when taking into account the remark from the last section on the rank of W_{QK}.



(a) Accuracy scaling law as d grows. The head dimension is fixed to 10 and the AoT has 20 heads.

(b) Scaling laws of the coefficients obtained by linear regression of the scaling laws on H as in figure 1.a.

Figure 2: Two different measurements of the constant $C(d, N)$. On the left when only d varies, and on the right when $d = d_h$ varies.

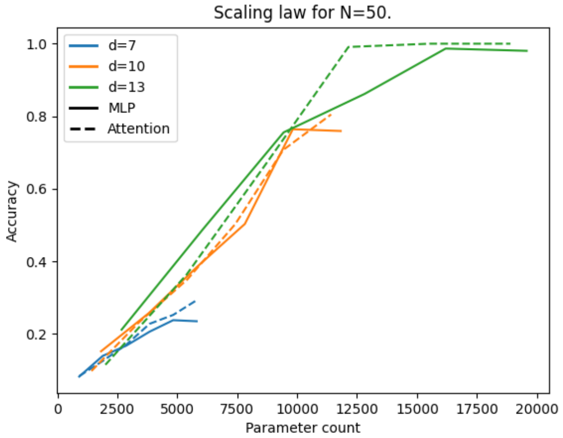


Figure 3: Accuracy scaling laws on the number of parameters for two models: an AoT and an MLP-based Transformer. Both of them have the same embedding dimension d.

The non-linearity σ is chosen to be the GELU function and is applied component-wise to its input vector. The MLP-based model needs at least one attention head in order to mix the tokens together.

Figure 3 shows that the scaling of the AoT and the MLP-based Transformer are the same for the same number of parameters. This means that the reason attention heads don't seem to remember associations in LLMs is not du to their memorization aptitude compared to MLPs. Why aren't attention heads found to remember association unlike MLPs ? Their could be two remaining reasons: either they are remembering associations but we didn't find it yet, or they are used to perform tasks that MLPs cannot do, like operation between residual stream, and thus give the remembering task to MLPs. Further experiments are needed to assess these hypotheses.

5 Upper bounds for sequence encoders

In Theorem 1, we are able to upper bound the divergence of the best AoT by the divergence of the best sequence encoder. In this section, we want to uppper bound the sequence encoder divergence to have better control on the performance of the best AoT.

This task is hard in general and is highly dependent on the fact that we are optimizing the KL divergence. For example, when a distribution π is close to uniform, it will be easily approximable by a sequence encoder f. Especially, the centered logit difference

$$\mathcal{Z}_{t_{1:S}}^{t_{S+1}} := f(t_{1:S})_{t_{S+1}} - \log(\pi(t_{S+1}|t_{1:S})) - \mathbb{E}_{t_{S+1}}[f(t_{1:S})_{t_{S+1}} - \log(\pi(t_{S+1}|t_{1:S}))]$$

will be close to 0. In this setting, doing a Taylor on the KL divergence reveals that the first order term will be the L_2 norm of $\mathcal{Z}_{t_{1:S}}^{t_{S+1}}$. In comparison, if we optimize for the total variation loss, one would find a different first order approximation.

In a different setting than the previous remark, we are able to control the divergence when the target probability satisfies Assumption 1. In that case, we can think of π as implementing a look-up table with some noise, and we define $g : [N]^S \to [N]$ the true look-up table meaning that $\pi(g(t_{1:S})|t_{1:S}) \simeq 1$. Althought Theorem 2 holds for any distribution π and function g, the upper bound is to be read with the above relation between π and g in mind.

Theorem 2. *Let $g : [N]^S \to [N]$. There exists $W \in \mathbb{R}^{d,N}$ such that*

$$||WW^T - I_d||_\infty \leq C = \sqrt{\frac{32\log(N+1)}{d}}$$

We choose $f_{W,E}$ with $E(t_{1:S}) = \lambda(t_{1:S})W_{g(t_{1:S})}^T$, and $\lambda(t_{1:S})$ the solution to equation (6).

$$-H(\pi_{t_{1:S}}) = \log\left(\sum_j e^{\lambda(t_{1:S})(W_j - W_{g(t_{1:S})})^T W_{g(t_{1:S})}}\right) \tag{6}$$

We obtain the bound

$$d_{KL}(\pi, \mathcal{L}(N,S,d)) \leq \mathbb{E}_{t_{1:S}}\left[(1-\pi(g(t_{1:S})|t_{1:S}))\log\left(\frac{N-1}{e^{-H(\pi_{t_{1:S}})}-1}\right)\left(\frac{1+2C+Cd_{TV}(\tilde{\pi}_{t_{1:S}}, \pi_{unif})}{1-2C}\right)\right]$$

Where, if $t_{S+1} \neq g(t_{1:S})$, $\pi_{unif}(t_{S+1}) = \frac{1}{N-1}$ and $\tilde{\pi}_{t_{1:S}}(t_{S+1}) = \frac{\pi_{t_{1:S}}(t_{S+1})}{1-\pi(g(t_{1:S})|t_{1:S})}$, and 0 otherwise for both distributions.

One can see that at the limit, the above bound converges to 0 for both the case $H(\pi_{t_{1:S}}) = 0$ meaning that the distribution is indeed a look-up table, or for $H(\pi_{t_{1:S}}) = -\log(N)$ meaning that the distribution is uniform. These cases corresponds respectively to $\lambda(t_{1:S}) \to +\infty$ or $\lambda(t_{1:S}) \to 0$. A simpler bound for Theorem 2 can be found by simplifying the expectation, and gives

$$d_{KL}(\pi, \mathcal{L}(N,S,d)) \leq \frac{1+4C}{1-2C}\mathbb{E}_{t_{1:S}}\left[\pi(t_{S+1} \neq g(t_{1:S})|t_{1:S})\right]\log\left(\frac{N-1}{e^{-\mathbb{E}_{t_{1:S}}[H(\pi_{t_{1:S}})]}-1}\right)$$

which only involves the average entropy and the average probability of the event $t_{S+1} \neq g(t_{1:S})$. The constant C comes from using the Johnson-Linderstrauss Lemma on the canonical basis. It may be improved since we only need to control the greatest positive coefficient of $WW^T - I_d$.

6 Related Work

Comparison with previous State-of-the-art: Our Theorem 1 is an improvement of the Theorem 1 in [9] in the case of language models. Indeed, the paper has the fundamental limitation that $S \leq d$, which is unrealistic for large language models[5]. This assumption is in part justified by their paper focusing on Vision Transformers (ViT), where keys and queries can come from different sets of vectors. In that case, this introduces a limitation depending on the rank of the family of queries Q: $T_0 \leq H(\min(Q, d_h) - 1) + 1$. And as stated in their Proposition 2, this bound is tight in case where the keys are shared by all examples, making the queries the only way to distinguish examples. Our result doesn't have such tightness since we choose a word and a positional embedding that will make the set of keys distinct for each exemples.

When the keys and queries are chosen to have full rank, their proof still has to limit to $S \leq d$, yet we don't. This is because they use the result from [1] stating that W_{QK}^h can be chosen to produce any attention pattern a if $S \leq d$. They then solve the linear system $q_{1:S}^t W_{QK} k_j = \log(a_{i,j})$ for each assocation. Rather than using the attention pattern's expressivity, our proof makes use of the whole attention layer's expressivity, and we are able to obtain the maximal memory capacity of Hd_h, and even $Hd_h + d$ if we use the skip connection, which is a strict improvement on their memory capacity of $H(d_h - 1) + 1$.

[5] In GPT2 $S = 1024$ while $d = 768$, yet in GPT3 $S = 2048$ and $d = 12288$. So this limitation depends on the use-case made by the language model, but the tendency in industry is to have $S > d$, with most recent context window in the million tokens for Gemini 1.5 Pro.

Small-width large-depth MLP-based Transformers: Another result on the memorization capacity of Transformer is given by [8] where they leverage the depth of Transformers. Their setup and architecture is different from ours, making use of the MLP layers and predicting not only the last token but a complete sequence. Still, their Theorem 4.1 is closest to ours in that they predict the same token at every position, and we can adapt their result to allow a fair comparison. They are able to memorize T_0 examples using $O\left(S + N + \sqrt{T_0 \log(T_0)}\right)$ parameters[6]. To do so, they use lots of layers which each have a constant number of parameters. In the parameter counts, $O\left(\sqrt{T_0 \log(T_0)}\right)$ parameters come from the MLP layers, which is the majority, and the attention layers, although crucial, need only $O(S)$ parameters[7].

As a comparison, our model in Corollary 1 scales as $2S + 4N + 8T_0$ parameters. So, in terms of scaling, combining MLPs and attention layers with depth is much better.

Attention simulating MLPs: Since the MLP memorization properties are well known, one could use attention heads to memorize by simulating MLPs. [6] proposes a method to simulate MLP layers using attention heads. They prove that an attention head can simulate an MLP neuron if it can use an appropriate attention mask. If the head has to learn the mask, then the construction still uses one attention per neuron, but each head needs to satisfy $d_h \geq 1 + S$ and $d \geq 2 + S$. This makes this architecture memorize exactly T_0 associations using $O((d + S)(S + N + T_0))$ parameters.

If exact memorization is strictly worse than the previous architecture, combining this idea with the scaling of Theorem 2 in [3] for the approximate memorization in MLPs means that the number of parameters could be competitive using $O\left(\log\left(\frac{1}{\varepsilon}\right)\frac{T_\varepsilon \log(T_\varepsilon)}{O\left(\log\left(\frac{d}{\log(T_\varepsilon)}\right)\right)}(1 + \frac{S}{d}) + (d + S)(S + N)\right)$ to have an ε-approximation. Theorem 1's parameter count is $O(d(S + N + T_\varepsilon - d))$, so using attention to simulate MLPs is competitive when $S \log(N) \log\left(\frac{1}{\varepsilon}\right) = O(d \log(d))$.

Associative Memory: In [2, 4], the authors introduce a linear memorization framework for the association task. The learn association with the sequence encoder $W_U W E$ where $E : [N] \to \mathbb{R}^d$ is a fixed random encoder, $W \in \mathbb{R}^{d,d}$ is learnable and $W_U \in \mathbb{R}^{N,d}$ outputs logits and is fixed random. In this setting, they show that using the d^2 learnable parameters from W, the model memorizes d associations. This coincides with our finding when $H = 0$, although we use the word embedding and output unembedding.

7 Conclusion

In this paper, we have advanced the state of the art in exact memorization by addressing and overcoming the limitation that the sequence size must be smaller than the embedding dimension. Through empirical analysis, we computed scaling laws based on our Corollary 1, confirming that the scaling with respect to H is linear. However, for d_h, the scaling appears more complex, showing a dependence on d_h^2, meaning that the best performances per parameter is acheived for $d = d_h$. In that case, we found the scaling to be $C(N)d^3H$.

Contrary to our initial hypothesis that Attention heads would perform worse than MLPs in terms of memorization, we were surprised to find very similar scaling laws for both mechanisms. Additionally, we introduced the concept of memorization in distribution and derived an upper bound on the KL-divergence. Further progress in this area will require a deeper understanding of the expressivity of the attention mechanism, a topic we have begun exploring in this work.

References

[1] Srinadh Bhojanapalli, Chulhee Yun, Ankit Singh Rawat, Sashank Reddi, and Sanjiv Kumar. Low-rank bottleneck in multi-head attention models. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 864–873. PMLR, 13–18 Jul 2020.

[6] Looking at their Definition 3.1, one could wonder if the result hides a constraint on the embedding dimension. This is true for their general statement, and using packing number one can obtain a lower bound that d must satisfy. However, in our setting, we work with tokens rather than vectors, and, in that case, their result loses the constraint by using a word embedding. This can be seen in their proof by acknowledging that A.1 is trivial using a word embedding, and that this only impacts the constant R in A.6 minimally.

[7] See their Remark 3.7

[2] Alberto Bietti, Vivien Cabannes, Diane Bouchacourt, Herve Jegou, and Leon Bottou. Birth of a transformer: A memory viewpoint. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[3] Sébastien Bubeck, Ronen Eldan, Yin Tat Lee, and Dan Mikulincer. Network size and size of the weights in memorization with two-layers neural networks. In *Neural Information Processing Systems*, 2020.

[4] Vivien Cabannes, Elvis Dohmatob, and Alberto Bietti. Scaling laws for associative memories. In *The Twelfth International Conference on Learning Representations*, 2024.

[5] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

[6] Robert Huben and Valerie Morris. Attention-only transformers and implementing mlps with attention heads, 2023.

[7] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online, November 2020. Association for Computational Linguistics.

[8] Junghwan Kim, Michelle Kim, and Barzan Mozafari. Provable memorization capacity of transformers. In *The Eleventh International Conference on Learning Representations*, 2023.

[9] Sadegh Mahdavi, Renjie Liao, and Christos Thrampoulidis. Memorization capacity of multihead attention in transformers. In *The Twelfth International Conference on Learning Representations*, 2024.

[10] Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[11] Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, 2023.

[12] Neel Nanda, Senthooran Rajamanoharan, János Kramár, and Rohin Shah. Fact Finding: Attempting to Reverse-Engineer Factual Recall on the Neuron Level, December 2023.

[13] Adam Roberts, Colin Raffel, and Noam Shazeer. How Much Knowledge Can You Pack Into the Parameters of a Language Model?, October 2020. arXiv:2002.08910 [cs, stat].

[14] Gal Vardi, Gilad Yehudai, and Ohad Shamir. On the optimal memorization power of reLU neural networks. In *International Conference on Learning Representations*, 2022.

[15] Alexandre Variengien and Eric Winsor. Look before you leap: A universal emergent decomposition of retrieval tasks in language models. In *ICML 2024 Workshop on Mechanistic Interpretability*, 2024.

[16] Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023.

A Proofs

A.1 Lemma 2

Before proving Lemma 2, let us state Lemma 3 below. We will prove it in appendix A.5.

Lemma 3. *For any $n \in \mathbb{N}^*$, there exists $x_1, ..., x_n > 0$ such that for any $P \in \mathbb{Z}[X_1, ..., X_n]^*$, $P(x_1, ..., x_n) \neq 0$.*

Let us prove this result in the case $T_0 \geq Hd$, since otherwise, one can add extra examples to prove the result and remove them later. Recall that throughout the proof $d \geq d_h$. Thanks to Lemma 3, there exists $(N+L)d$ positive numbers (X_i) such that for any non-zero polynomial with coefficient in $\mathbb{Z}$ of degree d on $(N+L)d$ variables, then the polynomial is non-zero when evaluated on X_i. We use these numbers to construct our word and positional embeddings by assigning each to a coefficient of our embeddings e and pos. We note $x(t,j) = e(t_j) + pos_j$, where t denotes a sequence of S tokens among the N^S possible sequences, and we define

$$f : W \to \left(\frac{\sum_j^S e^{x(t,S)^T W x(t,j)} x(t,j)}{\sum_j^S e^{x(t,S)^T W x(t,j)}} \right)_{t \in [T_0]} \in \mathbb{R}^{d, T_0}$$

Our goal is to choose matrices W_{QK}^i such that $A = [f(W_{QK}^1), ..., f(W_{QK}^H)] \in \mathbb{R}^{Hd, T_0}$ has rank Hd. This is equivalent to showing that its rows are free in $\mathbb{R}^{T_0}$. It is in fact sufficient to have the image of f not contained in any hyperplane of $\mathbb{R}^{d,T_0}$, which is the property we will prove. This way, we can iteratively concatenate to A a new $f(W_{QK}^i)$ that will increase the rank by d as long as the rank is lower than T_0.

Let $(v_t)_{t \in [T_0]}$ such that

$$\forall\, W \in \mathbb{R}^{d,d}, \text{rank}(W) = d_h, \sum_{t \in [T_0]} v_t^T f(W)_t = 0 \tag{7}$$

Let $i_{\max}(t) = \arg\max_i (x(t,i)_1)$ the indice of the greatest x on the first dimension. We can rewrite the equality (7) as

$$\begin{aligned} 0 &= \sum_{t \in [T_0]} \frac{\sum_i^S e^{x(t,S)^T W (x(t,i) - x(t, i_{\max}(t)))} v_t^T x(t,i)}{\sum_i^S e^{x(t,S)^T W (x(t,i) - x(t, i_{\max}(t)))}} \\ &= \sum_{t \in [T_0]} \sum_i^S e^{x(t,S)^T W (x(t,i) - x(t, i_{\max}(t)))} v_t^T x(t,i) \frac{1}{1 + \sum_{i \neq i_{max}(t)}^S e^{x(t,S)^T W (x(t,i) - x(t, i_{\max}(t)))}} \end{aligned} \tag{8}$$

Let us only consider for the rest of the proof matrices W such that $W_{i,j} = 0$ if $i \neq 1$ or $j \neq 1$, which has rank 1. In this case, we have the exponent $x(t,S)^T W (x(t,i) - x(t, i_{\max}(t)))$ equal to $x(t,S)_1^T W_{1,1} (x(t,i)_1 - x(t, i_{\max}(t))_1)$ so by definition of $i_{\max}(t)$, it is negative when $W_{1,1} = w$ is positive. Thus taking $w \to +\infty$ makes every exponential but the index $i_{\max}(t)$ converge to 0. For a large enough w we can use the Taylor expansion of $x \to \frac{1}{1+x}$ around valid on $[0,1]$.

$$\frac{1}{1 + \sum_{i \neq i_{\max}(t)}^S e^{x(t,S)_1^T w (x(t,i)_1 - x(t, i_{\max}(t))_1)}} = \sum_{j=0}^{+\infty} \left(- \sum_{i \neq i_{\max}(t)}^S e^{x(t,S)_1^T w (x(t,i)_1 - x(t, i_{\max})_1)} \right)^j \tag{9}$$

Finally, we let $\mathcal{S}_j(t) = \{(i_1, ..., i_j) \in [S]^j | \forall k \in [1,j], i_k \neq i_{\max}(t)\}$ that enumerates all indices for the product of j sums in (9). Developping these product of sums into a sum of products transforms the term into,

$$\sum_{j=0}^{+\infty} \sum_{i_1, ..., i_j \in \mathcal{S}(t)} (-1)^j e^{x(t,S)_1^T w \sum_{k=1}^j x(t,i_k)_1 - x(t, i_{max}(t)_1)} \tag{10}$$

Finally, we (10) back in (8) to have the following equality for all w sufficiently large.

$$\sum_{t \in [T_0]} \sum_{j=0}^{+\infty} \sum_{i_0=1}^{S} \sum_{i_1, ..., i_j \in \mathcal{S}(t)} (-1)^j v_t^T x(t, i_0) e^{x(t,S)_1^T w \sum_{k=0}^j x(t,i_k)_1 - x(t, i_{max}(t)_1)} = 0 \tag{11}$$

Equality (11) is a functional equality, were the functions are exponential with different exponents. Since the family of exponential function is free, we have that coefficient in front of each different exponent is 0. The next step in the proof is to find a sequence of indices such that the exponent is almost unique to that sequence of indices and to that token sequence.

Let $t \in [T_0]$ and consider any $n \in [2, d+1]$, with the exponent generated by taking n^i times the indice $i \neq i_{\max}(t)$ in the sum in (12). The crafted exponent is $x(t,S)_1 w \sum_{i \neq i_{\max}(t)} n^i (x(t,i)_1 - x(t, i_{\max}(t))_1)$. Now let t' another token sequence and $c(t)_i$ other coefficient for the number of time that the indice i was chosen, such that both exponents are equal

$$x(t,S)_1 \sum_{i \neq i_{\max}(t)} n^i (x(t,i)_1 - x(t, i_{\max}(t))_1) = x(t',S)_1 \sum_{i \neq i_{\max}(t')} c(t')_i (x(t',i)_1 - x(t', i_{\max}(t'))_1) \quad (12)$$

We will prove that this implies that $t = t'$ and $c(t')_i = n^i$. Since (12) acts as a polynomial equation on the embedding coefficients, we can use the property of the embedding in Lemma 3 to identify the coefficients on each side. First let us start with the positional embeddings. We only have once the coefficients $pos_S^T pos_i$ on each side, so we can identify each. This means that $i_{\max}(t) = i_{\max}(t')$, by identifying the negative positional embeddings, and that $c(t')_i = n^i$. To identify the token sequences, we can start by looking at $e(t_S)_1 * \left(\sum_{i \neq i_{\max}(t)}^{S} n^i (pos_{i,1} - pos_{i_{\max}(t),1}) \right)$ and compare it to its counterpart, this gives us $e(t_S) = e(t'_S)$ meaning $t_S = t'_S$. Then we remove the extra information we already deduced and the equality (12) becomes

$$\sum_{i \neq i_{\max}(t)}^{S} n^i e\left(t_{1:S}\right)_1 = \sum_{i \neq i_{\max}(t)}^{S} n^i e\left(t'_i\right)_1 \quad (13)$$

We now apply Lemma 3 to (13), so there need to be as many $e(q)$, $q \in [N]$ on each side. Since for q the number of $e(q)$ can be written uniquely in base n on each side, it means that $t_{1:S} = q \iff t'_i = q$. If the sum in (13) is empty on both side, which it can be, then this means that $t_i = t_{i_{max(t)}}$ for all i, and likewise for t', and since $t_S = t'_S$, we have $t = t'$.

This shows that, for an exponent of the previous form to be equal to another one, both need to have the same number of each indices, they must come from the same token sequence, and their number of indices differ by at most one, depending on whether $i_0 = i_{\max}(t)$ or not. So, we can write the coefficient in front of that exponential as

$$v_t^T \sum_{i=1}^{N} x(t,i) N_i^n = 0$$

where N_i^n is the number of permutation of a sequence containing $n^k - \mathbb{1}_{i=k \cap i_{\max}(t) \neq k}$ times the indice k and starting with indice i. The system writes as

$$v_t^T \mathcal{N} X(t) = 0$$

with $X(t)_i = x(t,i)$ and $\mathcal{N}_{n,i} = N_i^n$. The matrix $\mathcal{N} X(t)$ has non-zero determinant by property of Lemma 3, so the system's only solution is for $v_t = 0$.

A.2 Theorem 1

In what follows we work under Assumption 2, and we let $f_{W,E}$ that achieves the minimum distance $d_{KL}(\pi, \mathcal{L}(N,S,d))$, thanks to Lemma 1.

Let us first prove the theorem when $\varepsilon = 0$. Observe that the skip connection contribution can be expressed as a type of attention head. Indeed, for any matrices $W_{OV} \in \mathbb{R}^{d,d}$ with full rank, and we define the new embeddings as $e'(t_s) = W_{OV}^{-1} e(t_s)$ and $pos'_s = W_{OV}^{-1} pos_s$. Now, if we let $W_{QK}^{\lambda} = \lambda I_d$, we obtain in the limit that $x(t,i) = W_{OV} \sum_{s=1}^{S} a(t)_s x'(t,i) + o(e^{-\lambda\alpha})$ as in (1), with some $\alpha > 0$. This attention head has inner dimension d.
We take $H = \left\lceil \frac{T_0 - d}{d_h} \right\rceil$ attention modules. We have $\mathcal{T}(t_{1:S}) = W_U W_O W_V A'(t_{1:S}) + o(e^{-\lambda\alpha})$ where A' is the attention before output augmented by the "fake" head from the skip connection. We can now apply Lemma 2 to A', for some choice of e, pos and W_{QK}^h such that A' has rank T_0. After multiplying by W_V with full rank, the rank stays T_0. Finally, we take $W_U = W$, and W_O that solves the system of T_0 equations $E = W_O W_V A'$. We end-up with $\mathcal{T} = f_{W,E} + o(e^{-\lambda\alpha})$.

Now, let $\varepsilon > 0$, and T_ε the smallest number of sentences whose cumulative probability is greater than $1 - \varepsilon$. We have,

$$\begin{aligned}
|d_{KL}(\pi, f_{W,E}) - d_{KL}(\pi, \mathcal{T})| &= \left| \mathbb{E}_{t_{1:S}} \left[\log \left(\frac{\mathbb{E}_{t_{S+1}} \left[e^{\mathcal{T}(t_{1:S})_{t_{S+1}} - \mathbb{E}_{t_{S+1}}[\mathcal{T}(t_{1:S})_{t_{S+1}}]} \right]}{\mathbb{E}_{t_{S+1}} \left[e^{f_W(t_{1:S})_{t_{S+1}} - \mathbb{E}_{t_{S+1}}[f_W(t_{1:S})_{t_{S+1}}]} \right]} \right) \right] \right| \\
&\leq \mathbb{E}_{t_{1:S}} \left[||\mathcal{T}(t_{1:S}) - f_W(t_{1:S}) - \mathbb{1}\mathbb{E}_{t_{S+1}}[\mathcal{T}(t_{1:S})_{t_{S+1}} - f_W(t_{1:S})_{t_{S+1}}]||_\infty \right] \\
&\leq \mathbb{E}_{t_{1:S}} \left[||(I_d - \mathbb{1}\Pi_{t_{1:S}}^T)(\mathcal{T}(t_{1:S}) - f_{W,E}(t_{1:S}))||_\infty \right] \\
&\leq \mathbb{E}_{t_{1:S}} \left[||(I_d - \mathbb{1}\Pi_{t_{1:S}}^T) W_U (W_O W_V A' - E) t_{1:S}||_\infty \right] \\
&\leq \mathbb{E}_{t_{1:S}} \left[||I_d - \mathbb{1}\Pi_{t_{1:S}}^T||_{2,\infty} ||W_U (W_O W_V A' - E) t_{1:S}||_2 \right] \\
&\leq 2\mathbb{E}_{t_{1:S}} \left[||W_U (W_O W_V A' - E) t_{1:S}||_2 \right] + o(e^{-\lambda\alpha})
\end{aligned} \tag{14}$$

Let $P_1^T P_1$ the projection onto the T_ε most likely sentences and S_1 the set of those sentences, meaning that for $t_{1:S} \in S_1$, $P_1^T P_1 t_{1:S} = t_{1:S}$. Let $P_2^T P_2$ and S_2 for the rest of the sentences. We have $I_d = P_1^T P_1 + P_2^T P_2$. Using the singular value decomposition of $W_V A' = U I_\Sigma P_1 V$, we take $W_O = E P_1^T (P_1 V P_1^T)^{-1} I_\Sigma^{-1} U^T$, and we have that $(W_O W_V A - E) P_1^T = 0$. We also take $W_U = W$ as before.

$$\begin{aligned}
\mathbb{E}_{t_{1:S}} \left[||W(W_O W_V A - E) t_{1:S}||_2 \right] &= \mathbb{E}_{t_{1:S}} \left[||W(W_O W_V A - E) P_2^T P_2 t_{1:S}||_2 \right] \\
&= \mathbb{E}_{t_{1:S}} \left[||WE(P_1^T (P_1 V P_1^T)^{-1} P_1 V - I_d) P_2^T P_2 t_{1:S}||_2 \right] \\
&\leq ||WE||_2 \mathbb{E}_{t_{1:S}} \left[||(P_1^T (P_1 V P_1^T)^{-1} P_1 V - I_d) P_2^T P_2 t_{1:S}||_2 \right] \\
&\leq \varepsilon ||WE||_2 \sqrt{1 + ||P_1^T (P_1 V P_1^T)^{-1} P_1 V||_2^2} \\
&\leq \varepsilon ||WE||_2 \sqrt{1 + ||(P_1 V P_1^T)^{-1}||_2^2}
\end{aligned} \tag{15}$$

Thus, there exists matrices e, pos, W_O^p, W_V^p, and W_{QK}^p such that

$$|d_{KL}(\pi, f_{W,E}) - d_{KL}(\pi, \mathcal{T})| \leq C\varepsilon ||WE||_2 + o(e^{-\lambda\alpha})$$

A.3 Theorem 2

Recall from the theorem that $f(t_{1:S}) = f_{W,E}(t_{1:S}) = \lambda(t_{1:S}) W W_{g(t_{1:S})}^T$, so by definition of C, one has $f(t_{1:S})_j \leq C$ if $j \neq g(t_{1:S})$, and $f(t_{1:S})_{g(t_{1:S})} \leq 1 + C$.

$$\begin{aligned}
d_{KL}(\pi, f) &= \mathbb{E}_{t_{1:S}}[H(\pi_{t_{1:S}})] - \mathbb{E}_{t_{1:S}, t_{S+1}} \left[\log \left(\frac{f_{t_{S+1}}(t_{1:S})}{\sum_j e^{f_j(t_{1:S})}} \right) \right] \\
&= \mathbb{E}_{t_{1:S}}[H(\pi_{t_{1:S}})] + \mathbb{E}_{t_{1:S}, t_{S+1}} \left[\log \left(\sum_j e^{f_j(t_{1:S}) - f_{t_{S+1}}(t_{1:S})} \right) \right] \\
&= \mathbb{E}_{t_{1:S}}[H(\pi_{t_{1:S}})] + \mathbb{E}_{t_{1:S}} \left[\log \left(\sum_j e^{f_j(t_{1:S}) - f_{g(t_{1:S})}(t_{1:S})} \right) \right] \\
&+ \mathbb{E}_{t_{1:S}, t_{S+1}}[f_{g(t_{1:S})}(t_{1:S}) - f_{t_{S+1}}(t_{1:S})] \\
&= \mathbb{E}_{t_{1:S}}[H(\pi_{t_{1:S}})] + \mathbb{E}_{t_{1:S}} \left[\log \left(\sum_j e^{\lambda(t_{1:S})(W_j - W_{g(t_{1:S})})^T W_{g(t_{1:S})}} \right) \right] \\
&+ \mathbb{E}_{t_{1:S}, t_{S+1}}[f_{g(t_{1:S})}(t_{1:S}) - f_{t_{S+1}}(t_{1:S})]
\end{aligned}$$

For each $t_{1:S}$, $H(\pi_{t_{1:S}}) \in [0, \log(N)]$ and

$$\lambda \to \log \left(\sum_j e^{\lambda(t_{1:S})(W_j - W_{g(t_{1:S})})^T W_{g(t_{1:S})}} \right)$$

is decreasing from $\log(N)$ to 0. Thus there exists a solution $\lambda(t_{1:S})$ to equation (6). Moreover since $W_{t_{S+1}}W^T_{t'_{S+1}} \leq C$ for $t'_{S+1} \neq t_{S+1}$ and $||W_{g(t_{1:S})}||_2^2 \geq 1 - C$, we obtain the following bound on λ,

$$\lambda(t_{1:S}) \leq \frac{1}{1-2C} \log\left(\frac{N-1}{e^{-H(\pi_{t_{1:S}})} - 1}\right)$$

Taking $\lambda(t_{1:S})$ to be this solution leaves us with,

$$\begin{aligned}
d_{KL}(\pi, f) &= \mathbb{E}_{t_{1:S}, t_{S+1}}[f(t_{1:S})_{g(t_{1:S})} - f(t_{1:S})_{t_{S+1}}] \\
&= \sum_{t_{1:S}} \pi(t_{1:S}) f(t_{1:S})_{g(t_{1:S})} - \sum_{t_{1:S}, t_{S+1}} \pi(t_{1:S})\pi(t_{S+1}|t_{1:S}) f(t_{1:S})_{t_{S+1}} \\
&= \sum_{t_{1:S}} \pi(t_{1:S})(1 - \pi(g(t_{1:S})|t_{1:S})) f(t_{1:S})_{g(t_{1:S})} - \sum_{t_{S+1} \neq g(t_{1:S})} \pi(t_{1:S})\pi(t_{S+1}|t_{1:S}) f(t_{1:S})_{t_{S+1}} \\
&= \sum_{t_{1:S}} \pi(t_{1:S})(1 - \pi(g(t_{1:S})|t_{1:S})) \left(f(t_{1:S})_{g(t_{1:S})} + \frac{1}{N-1} \sum_{t_{S+1}} f(t_{1:S})_{t_{S+1}} \right) \\
&- \sum_{t_{1:S}, t_{S+1}} \pi(t_{1:S}) \left(\pi(t_{S+1}|t_{1:S}) - \frac{1}{N-1}(1 - \pi(g(t_{1:S})|t_{1:S})) \right) f(t_{1:S})_{t_{S+1}} \\
&\leq \sum_{t_{1:S}} \pi(t_{1:S})(1 - \pi(g(t_{1:S})|t_{1:S})) \lambda(t_{1:S}) \left(1 + 2C + C d_{TV}(\tilde{\pi}_{t_{1:S}}, \pi_{\text{unif}})\right) \\
&\leq \mathbb{E}_{t_{1:S}} \left[(1 - \pi(g(t_{1:S})|t_{1:S})) \log\left(\frac{N-1}{e^{-H(\pi_{t_{1:S}})} - 1}\right) \left(\frac{1 + 2C + C d_{TV}(\tilde{\pi}_{t_{1:S}}, \pi_{\text{unif}})}{1 - 2C}\right) \right]
\end{aligned}$$

A.4 Theorem 3

In this appendix, our goal is to formalize the intuition from section 4.1 on how to get rid of the assumption on π in Theorem 1. The main problem with no assummption on π is that the weights of the sequence encoder that best fit π goes to infinity. The rate at which it goes to infinity is important to control the term $\varepsilon ||W_\varepsilon E_\varepsilon||_2$ in Theorem 1. Thus, we need to approximate π by another distribution, which we know we can approximate slowly.

Definition 2. *A distribution π is called* ***polynomialy approximable*** *if there exist a sequence of sequence encoders f_δ and constants $c > 0, \alpha \geq 1$ such that for every token sequence t such that $\pi(t_{S+1}|t_{1:S}) > 0$, we have*

$$c\delta^\alpha \leq \left| \log\left(\frac{\pi(t_{S+1}|t_{1:S})}{\mathit{softmax}(f_\delta(t_{1:S}))_{t_{S+1}}}\right) \right| \leq \delta \tag{16}$$

We denote $\mathcal{M}(N, S, d)$ the set of all slowly approximable distribution, and

$$d_{KL}(\pi, \mathcal{M}(N, S, d)) := \inf_{\phi \in \mathcal{M}(N,S,d)} d_{KL}(\pi, \phi)$$

With equation (16), we can control nicely the growth of the norm of the sequence of sequence encoder f_δ that approximate the distribution, using the following Lemma.

Lemma 4. *Let π, $c > 0$ and f_δ that satisfy equation (16) when $\pi(t_{S+1}|t_{1:S}) > 0$. Then, for all $t_{1:S}$, $\max_i f_\delta(t_{1:S})_i - \min_i f_\delta(t_{1:S})_i = O\left(\log\left(\frac{1}{\delta}\right)\right)$.*

Lemma 4 states that as long as the sequence encoders approximate the distribution at a speed at most polynomial in δ, then the norm will not be too large. If the speed in the lower bound was $\delta^{\frac{1}{\delta}}$, the resulting norm would be $O\left(\frac{1}{\delta}\log\left(\frac{1}{\delta}\right)\right)$, which would diverge too fast for Theorem 1. In equation (16), one can always suppose that the right hand-side is satisfied by re-indexing the sequence f_δ. **Whether any distribution which is the limit of a sequence encoder is in the closure of $\mathcal{M}(N, S, d)$ is an open question**. Still, we can provide an upper bound like Theorem 1, but this time for all π, using $\mathcal{M}(N, S, d)$ instead of $\mathcal{L}(N, S, d)$.

Theorem 3. *Let $\varepsilon > 0$. There exist an AoT $\mathcal{T}^*$ with embedding dimension d, head dimension d_h, and H attention heads, satisfying $d_h H + d \geq T_\epsilon$, such that*

$$d_{KL}(\pi, \mathcal{T}^*) \leq d_{KL}(\pi, \mathcal{M}(N, S, d)) + O\left(\varepsilon \log\left(\frac{1}{\varepsilon}\right)\right) \tag{17}$$

$\mathcal{T}^*$ has $d(S + 2N + 4d_h H)$ parameters.

Proof. Theorem 3. Let π_0 a distribution that is slowly approximable and f_δ a sequence of sequence encoders satysfying equation (16) for $c > 0$ and $\alpha \geq 1$. First, using the right hand-side of (16), we have

$$\begin{aligned} |d_{KL}(\pi, f_\delta) - d_{KL}(\pi, \pi_0)| &\leq \left| \mathbb{E}_{t_{1:S}, t_{S+1}} \left[\log \left(\frac{\pi_0(t_{S+1}|t_{1:S})}{\text{softmax}(f_\delta(t_{1:S}))_{t_{S+1}}} \right) \right] \right| \\ &\leq \mathbb{E}_{t_{1:S}} \left[\sum_{t_{S+1}=1}^{N} \pi(t_{S+1}|t_{1:S}) \left| \log \left(\frac{\pi_0(t_{S+1}|t_{1:S})}{\text{softmax}(f_\delta(t_{1:S}))_i} \right) \right| \right] \\ &\leq \delta \end{aligned} \tag{18}$$

since we apply (16) if $\pi(i|t_{1:S}) > 0$, and the majoration is otherwise trivial. Now, we use the bound from Theorem 1 in equation (14) and (15), but we don't majorate $||I_d - \mathbb{1}\Pi_{t_{1:S}}^T||_{2,\infty}$.

$$\begin{aligned} |d_{KL}(\pi, \mathcal{T}) - d_{KL}(\pi, f_\delta)| &\leq \varepsilon C \max_{t_{1:S}} ||(I_d - \mathbb{1}\Pi_{t_{1:S}}^T) f_\delta||_\infty \\ &\leq \varepsilon C \max_{t_{1:S}} \max_{||x||_\infty = 1} \begin{cases} \max_{t_{S+1}} f_\delta(x)_{t_{S+1}} - \mathbb{E}_{t_{S+1}}[f_\delta(x)_{t_{S+1}}] \\ \mathbb{E}_{t_{S+1}}[f_\delta(x)_{t_{S+1}}] - \min_{t_{S+1}} f_\delta(x)_{t_{S+1}} \end{cases} \\ &\leq \varepsilon C \max_{||x||_\infty = 1} (\max_{t_{S+1}} f_\delta(x)_{t_{S+1}} - \min_{t_{S+1}} f_\delta(x)_{t_{S+1}}) \\ &\leq \varepsilon C \max_{t_{1:S}} (\max_{t_{S+1}} f_\delta(t_{1:S})_{t_{S+1}} - \min_{t_{S+1}} f_\delta(t_{1:S})_{t_{S+1}}) \\ &\leq O\left(\varepsilon \log \left(\frac{1}{\delta} \right) \right) \end{aligned} \tag{19}$$

by using Lemma 4 for the last row. We conclude the proof by assembling (18) and (19) with $\varepsilon = \delta$, and by taking the infinimum over all polynomialy approximable distributions.

$$\begin{aligned} d_{KL}(\pi, \mathcal{T}^*) &= d_{KL}(\pi, \mathcal{T}^*) - d_{KL}(\pi, f_\delta) + d_{KL}(\pi, f_\delta) - d_{KL}(\pi, \pi_0) + d_{KL}(\pi, \pi_0) \\ &\leq O\left(\varepsilon \log \left(\frac{1}{\varepsilon} \right) \right) + \varepsilon + d_{KL}(\pi, \pi_0) \\ &\leq O\left(\varepsilon \log \left(\frac{1}{\varepsilon} \right) \right) + d_{KL}(\pi, \mathcal{M}(N, S, d)) \end{aligned} \tag{20}$$

□

Proof. Lemma 4. Let π, $c > 0$, $\alpha \geq 1$ and f_δ satisfying (16). Let $t_{1:S}$ and t_{S+1} such that $\pi(t_{S+1}|t_{1:S}) > 0$. We can write the left hand-side of (16) as:

$$\frac{e^{f_\delta(t_{1:S})_{t_{S+1}}}}{\sum_j e^{f_\delta(t_{1:S})_j}} \leq \pi(t_{S+1}|t_{1:S}) e^{-c\delta^\alpha} \text{ or } \frac{e^{f_\delta(t_{1:S})_{t_{S+1}}}}{\sum_j e^{f_\delta(t_{1:S})_j}} \geq \pi(t_{S+1}|t_{1:S}) e^{c\delta^\alpha} \tag{21}$$

Suppose that all choice of t_{S+1} end up in the right hand-side of (21), then, by summing this inequality for all t_{S+1} such that $\pi(t_{S+1}|t_{1:S}) > 0$, we get

$$1 \geq \sum_k \frac{e^{f_\delta(t_{1:S})_k}}{\sum_j e^{f_\delta(t_{1:S})_j}} \geq e^{c\delta^\alpha} > 1$$

which is a contraction. Thus, there exist a token t_{S+1} such that the left hand-side is true. We now keep this token for the rest of the proof. Since $\frac{e^{f_\delta(t_{1:S})_{t_{S+1}}}}{\sum_j e^{f_\delta(t_{1:S})_j}} \underset{\delta \to 0}{\longrightarrow} \pi(t_{S+1}|t_{1:S}) > 0$, then $e^{f_\delta(t_{1:S})_j - f_\delta(t_{1:S})_{t_{S+1}}} = O(1)$ for any other j satisfying $\pi(j|t_{1:S}) > 0$. This is because, all the logit that have non-zero limit probability after renormalization must grow at the same speed. It means that $f_\delta(t_{1:S})_{t_{S+1}} - f_\delta(t_{1:S})_j = \log(\pi(t_{S+1}|t_{1:S})) - \log(\pi(j|t_{1:S})) + o(1)$. One can do the same reasoning on the logits $f_\delta(t_{1:S}) - \min_j f_\delta(t_{1:S})_j$, meaning that all logits are positive. In this case, when $\pi(j|t_{1:S}) = 0$, we have at the limit that $0 \leq \frac{f_\delta(t_{1:S})_j - \min f_\delta(t_{1:S})}{f_\delta(t_{1:S})_{t_{S+1}} - \min f_\delta(t_{1:S})} < 1$ since the exponentials need in these cases to grow slower than the one of t_{S+1}. We denote this limit α_j. We can now write again the inequality (21),

$$
\begin{aligned}
\pi(t_{S+1}|t_{1:S})e^{-\frac{\delta}{c}} &\geq \frac{e^{f_\delta(t_{1:S})_{t_{S+1}}}}{\sum_j e^{f_\delta(t_{1:S})_j}} \\
&= \frac{\pi(t_{S+1}|t_{1:S})e^{o(1)}}{\sum_{\pi(j|t_{1:S})>0} \pi(j|t_{1:S})e^{o(1)} + \sum_{\pi(j|t_{1:S})>0} e^{(\alpha_j-1)(f_\delta(t_{1:S})_{t_{S+1}} - \min f_\delta(t_{1:S}))(1+o(1))}} \\
&\geq \frac{\pi(t_{S+1}|t_{1:S}) + o(1)}{1 + o(1) + (N-1)e^{(\max_j \alpha_j - 1)(\max f_\delta(t_{1:S}) - \min f_\delta(t_{1:S}))(1+o(1))}}
\end{aligned}
\tag{22}
$$

And finally, equation (22) for all tokens sequences gives the bound

$$
\max_{t_{1:S}}(\max f_\delta(t_{1:S}) - \min f_\delta(t_{1:S})) \leq \frac{\log\left(e^{\frac{\delta}{c}} - 1\right) - \log(N-1)}{1 - \max_j \alpha_j}(1 + o(1)) = O\left(\log\left(\frac{1}{\delta}\right)\right)
$$

□

A.5 Other results and proofs

Proof. Proposition 3: Since we are under Assumption 1, there exist $g : [N]^S \to [N]$ such that $\pi(g(t_{1:S})|t_{1:S}) = 1$. Let $W^t_{1:S} = \left(\cos\left(\frac{2\pi i}{N}\right), \sin\left(\frac{2\pi i}{N}\right)\right)$, and $E(t_{1:S}) = W^T_{g(t_{1:S})}$. Let $f_\lambda = \lambda W E$. Since π has 0 entropy for any of its conditional distribution, we have

$$
\begin{aligned}
d_{KL}(\pi, f_\lambda) &= \mathbb{E}_{t_{1:S}}\left[\log\left(\sum_j e^{f_\lambda(t_{1:S})_j - f_\lambda(t_{1:S})_{g(t_{1:S})}}\right)\right] \\
&= \mathbb{E}_{t_{1:S}}\left[\log\left(1 + \sum_{j \neq g(t_{1:S})} e^{\lambda \cos\left(\frac{2\pi}{N}(j - g(t_{1:S}))\right) - \lambda}\right)\right] \\
&= \mathbb{E}_{t_{1:S}}\left[\log\left(1 + \sum_{j \neq g(t_{1:S})} e^{-\lambda\left(1 - \cos\left(\frac{2\pi}{N}(j - g(t_{1:S}))\right)\right)}\right)\right]
\end{aligned}
$$

using trigonometric properties. Thus we have that, $d_{KL}(\pi, f_\lambda) \underset{\lambda \to +\infty}{\to} 0$. □

Proof. Lemma 1: Under Assumption 2, for $\varepsilon, \delta > 0$, we take f_ε such that

$$
|d_{KL}(\pi, f_\varepsilon) - d_{KL}(\pi, \mathcal{L}(N, S, d))| \leq \varepsilon.
$$

We can write the new bound

$$
|d_{KL}(\pi, f_\varepsilon) - d_{KL}(\pi, f_\delta)| = |\mathbb{E}_{t_{1:S}, t_{S+1}}[\log(\text{softmax}(f_\varepsilon(t_{1:S}))_{t_{S+1}}) - \log(\text{softmax}(f_\delta(t_{1:S}))_{t_{S+1}})]| \leq \varepsilon + \delta
$$

Since every probability $\pi(t_{1:S}, t_{S+1}) > 0$, we have that for every $t_{1:S}, t_{S+1}$, $\text{softmax}(f_\varepsilon(t_{1:S}))_{t_{S+1}})$ is bounded away from 0 and 1 for all ε. This implies that f_ε is bounded after centering it. So we can extract a converging subsequence from that centered sequence. To finish the proof, we simply need to show the closeness of $\mathcal{L}(N, S, d)$, but this is easily implied by the continuity of the constraint $\text{rank}(WE) \leq d$, which defines this set. □

Proof. Lemma 3: Let $n \in \mathbb{N}^*$. For each $P \in \mathbb{Z}[X_1, ..., X_n]^*$, $Ker(P) = \{x \in \mathbb{R}^n, P(x) = 0\}$ has 0 Lebesgue measure. This can be shown by taking $X \sim \mathcal{N}(0, I_d)$ who has density with respect to the Lebesgue measure and seeing that $P(X)$ still has density by standard properties of iid gaussian sum and product. This means that $P(X) = 0$ has probability 0, making $Ker(P)$ of measure 0. Now, since $\mathbb{Z}[X_1, ..., X_n]^*$ is countable, $\bigcup_{P \in \mathbb{Z}[X_1, ..., X_n]^*} Ker(P)$ still has measure 0. Thus there exist $x \in \bigcap_{P \in \mathbb{Z}[X_1, ..., X_n]^*} Ke\bar{r}(P)$. Now, these numbers can be chosen positive since the proof works the same way with the quadrant $\mathbb{R}^n_+$. □

B Experiments

We give here details on the experiments in section 4.2. The training procedure for every model was the following: generate a distribution π over 3 tokens, uniform over the first 2, and then one next-token was randomly chosen. This way, π satisfies Assumption 1. From π, we generated 1000 batches of 2^{10} elements each, and trained the model for 10 epochs. We used Adam with default parameters and a learning rate of 10^{-3}. Each model was trained twice on different seeds, and the accuracy was averaged. This github contains the code to reproduce the experiments using the notebook *experiments.ipynb*. All experiments were done on a single MacBook Air with an M2 chip and 16 Go of memory.

As explained earlier, we present here another scaling laws whose intend is to be a fair comparison with our Corollary 2. Exceptionaly, we used $N = 10$ to avoid training issues[8]. We train an AoT with $d = 2$, $d_h = 5$ and H from 1 to 20. Corollary 2 states that with $H = 20$, the model should be able to obtain exactly 1 accuracy. Now, how far is Corollary 2 from the empirical scaling ? Figure 4 shows that the true scaling at $d = 2$ seems to be around $1.7Hd_h$.

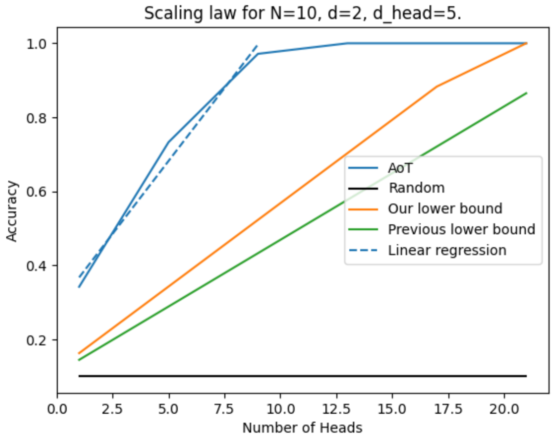


Figure 4

[8] The main issue with $d = 2$ is that the vector embedded in the plane cannot move freely since there exists repulsive interaction between vectors with different next-tokens. Thus, the accuracy becomes very dependant to the initialization. Taking a lower N solves the problem.